

**PENGONTROLAN SUHU MESIN MOBIL SECARA OTOMATIS DENGAN
MENGUNAKAN METODE KONTROLER PID**

SKRIPSI

TEKNIK ELEKTRO KONSENTRASI TEKNIK KONTROL

Diajukan untuk memenuhi persyaratan
memperoleh gelar Sarjana Teknik



MAHFUDHI IMANUDDIN SARASETYO

NIM. 125060305111003

UNIVERSITAS BRAWIJAYA

FAKULTAS TEKNIK

MALANG

2017

PERNYATAAN ORISINALITAS SKRIPSI

Saya menyatakan dengan sebenar-benarnya bahwa sepanjang pengetahuan saya dan berdasarkan hasil penelusuran berbagai karya ilmiah, gagasan dan masalah ilmiah yang diteliti dan diulas di dalam Naskah Skripsi/Tesis/Disertasi ini adalah asli dari pemikiran saya. tidak terdapat karya ilmiah yang pernah diajukan oleh orang lain untuk memperoleh gelar akademik di suatu Perguruan Tinggi, dan tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis dikutip dalam naskah ini dan disebutkan dalam sumber kutipan dan daftar pustaka.

Apabila ternyata di dalam naskah Skripsi/Tesis/Disertasi ini dapat dibuktikan terdapat unsur-unsur jiplakan, saya bersedia Skripsi/Tesis/Disertasi dibatalkan, serta diproses sesuai dengan peraturan perundang-undangan yang berlaku (UU No. 20 Tahun 2003, pasal 25 ayat 2 dan pasal 70).

Malang, 05 Juni 2017

Mahasiswa,

MAHFUDHI IMANUDDIN SARASETYO

NIM 125060305111003

RINGKASAN

Mahfudhi Imanuddin Sarasetyo, Jurusan Teknik Elektro, Fakultas Teknik Universitas Brawijaya, Mei 2017, Pengontrolan Suhu Mesin Mobil Dengan Menggunakan Kotroler PID, Dosen Pembimbing: Ir. Purwanto M.T. dan M. Aziz Muslim, S.T., M.T., Ph.D.

Salah satu masalah yang sering ditemui pada mobil adalah *Overheat* atau panas berlebih, terutama pada mobil yang sudah berumur. Gejala ini sering dialami Karena sistem pendinginan yang kurang maksimal. Sistem pendingin sendiri dibagi menjadi 2 yaitu sistem pendingin air (*Water Cooling System*) dan sistem pendingin udara (*Air Cooling System*). *Water Cooling System* menggunakan air sebagai media pendingin sedangkan *Air Cooling System* menggunakan udara sebagai media pendinginnya. *Water Cooling System* telah diaplikasikan pada semua mobil baik produksi lama maupun baru. Sedangkan mobil produksi tahun lama kebanyakan tidak dilengkapi dengan *Air Cooling System*. Untuk menjaga suhu kerja mesin mobil tetap pada suhu optimal pada berbagai macam kondisi maka diperlukan *Air Cooling System*.

Air Cooling System ini berupa pemasangan *Electric Fan*, *Electric Fan* ini akan mengalirkan udara ke radiator mesin mobil guna menjaga suhu optimal mesin pada kisaran 86°C. Penggunaan *electric fan* ini akan dipadu dengan kontroler PID metode *Ziegler-Nichols* agar mendapatkan hasil yang maksimal. Dengan menggunakan kontroler PID, sistem akan menghasilkan *error steady state* yang kecil dan akan mendapatkan settling time yang cepat. Proses perancangan kontroler PID dengan menggunakan metode Ziegler-Nichols didapatkan parameter PID dengan nilai parameter $K_p = 0.51$, $K_i = 0.002$, dan $K_d = 24.99$. Dari hasil pengujian kontroler PID didapatkan settling time sebesar 520 detik dan recovery time sebesar 118 detik.

Kata Kunci: Air Cooling System, Electric Fan, PID, Ziegler Nichols I

SUMMARY

Mahfudhi Imanuddin Sarasetyo, Department of Electrical Engineering, Faculty of Engineering University of Brawijaya, June 2017, Automatic Car Temperature Control Using PID Controller Method, Academic Supervisor: Ir. Purwanto M.T. and M. Aziz Muslim, S.T., M.T., Ph.D.

One of the problems that frequently found from old car was overheat or too much heat, this symptom frequently happens because some problems with the cooling system was not on the best performance. Cooling system it divided into two system which is Air Cooling System and Water Cooling System. Water cooling system uses water as a media to cooling down and Air Cooling System use air as a media to cooling down. Water cooling system has been applied to both old and new cars. But almost every old car did not equip with Air Cooling System. To keeping the working temperature of the car in any situation it need combination between Water Cooling System and Air Cooling System.

Air Cooling System is consisting of Electric Fan installation. Electric Fan will deliver air to the car radiator in purpose of keeping working temperature from the engine which is between 86°C. The electrical fan will using PID's controller, the system will gain small error steady state and get the fastest settling time. Design process that using the PID's controller with Ziegler-Nichols method obtained the PID's parameter such $K_p = 0.51$, $K_i = 0.002$, and $K_d = 24.99$. According to the result of the PID's controller experiments, 520 seconds are the settling time and 118 seconds are the recovery time.combine with PID Controller and Ziegler – Nichols Method

Keywords: *Air Cooling System, Electric Fan, PID, Ziegler Nichols I*

PENGANTAR

Bismillahirrohmanirrohim. Alhamdulillah, puji syukur penulis panjatkan kehadiran Allah SWT yang telah memberikan rahmat dan hidayah-Nya, sehingga penulis dapat menyelesaikan skripsi yang berjudul “Pengontrolan Suhu Mesin Mobil Secara Otomatis dengan Menggunakan Metode Kontrol PID” dengan baik. Tak lepas shalawat serta salam tercurahkan kepada junjungan kita Nabi Muhammad SAW yang telah menjadi suri tauladan bagi yang mengharapkan rahmat dan hidayah-Nya.

Penulis menyadari bahwa penyusunan skripsi ini tidak lepas dari bantuan, bimbingan serta dorongan dari berbagai pihak. Pada kesempatan ini penulis menyampaikan rasa terima kasih yang sebesar – besarnya kepada:

- Allah SWT yang telah memberikan kelancaran, kemudahan dan hidayah-Nya.
- Keluarga tercinta, kedua orang tua Judhi Indro Saraetyo dan Imami Faida yang selalu memberikan kasih sayang dan doanya yang tiada akhir. Kakak dan adik tercinta, dr. Nadhila Sharfina Pratiwi dan Azizah Khairina Sarasetyo yang selalu memberikan dukungan.
- Bapak M. Aziz Muslim, ST., MT., Ph.D. selaku Ketua Jurusan Teknik Elektro Universitas Brawijaya dan juga sebagai dosen pembimbing yang telah banyak memberikan kesempatan, nasehat, pengarahan, motivasi, saran dan masukan yang telah diberikan.
- Bapak Ir. Purwanto, M.T. selaku KKDK Sistem kontrol dan juga sebagai dosen pembimbing yang telah banyak memberikan kesempatan, nasehat, pengarahan, motivasi, saran dan masukan yang telah diberikan.
- Bapak Hadi Suyono, ST., MT., Ph.D. selaku Sekretaris Jurusan Teknik Elektro Universitas Brawijaya
- Bapak Ali Mustofa, ST., MT. selaku Ketua Program Studi S1 Jurusan Teknik Elektro Universitas Brawijaya
- Ardwin Novi Femisyah, terima kasih atas waktu, pengertian, semangat, bantuan dan kesabarannya yang telah diberikan.
- Sahabat-sahabat, Hanif, Nora, Zakiya, Suro, Faizal, Hanif, Gabriel, Tyo, dan Hesa terima kasih telah memberikan banyak bantuan, dukungan dan canda tawa.

- Keluarga besar Sistem Kontrol angkatan 2012 dan teman-teman angkatan 2012 “Voltage” atas do'a, semangat, serta dukungan yang diberikan pada penulis.
- Semua pihak yang telah memberikan bantuan serta dukungan baik secara langsung maupun tidak langsung atas penyusunan skripsi ini.

Dalam penyusunan skripsi ini, penulis menyadari bahwa skripsi ini belumlah sempurna, karena keterbatasan ilmu dan kendala–kendala lain yang terjadi selama pengerjaan skripsi ini. Oleh karena itu, penulis berharap kritik dan saran untuk penyempurnaan tulisan di masa yang akan datang. Semoga tulisan ini dapat bermanfaat dan dapat digunakan untuk pengembangan lebih lanjut.

Malang, 5 Juni 2017

Penulis

DAFTAR ISI

DAFTAR ISI.....	i
DAFTAR TABEL.....	ii
DAFTAR GAMBAR	v
DAFTAR LAMPIRAN.....	vi
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	3
1.3 Batasan Masalah.....	3
1.4 Tujuan Penelitian.....	3
1.5 Manfaat Penelitian.....	4
BAB II TINJAUAN PUSTAKA	5
2.1 Transportasi	5
2.2 Mesin Kendaraan Bermotor Roda 4.....	7
2.3 Sistem Pendingin Mobil.....	9
2.4 Gabungan Pendingin Air dan Pendingin Udara	19
2.5 Sensor Termokopel	20
2.5.1 Prinsip Kerja Termokopel.....	21
2.5.2 Temokopel Tipe K	22
2.6 Max 6675.....	23
2.7 Mesin <i>Direct Current</i> (DC).....	24
2.8 Electric Fan	26
2.9 Pulse with Modulation (PWM)	27
2.10 <i>High Current</i> 30 A DC Motor Driver dengan <i>Half-Bridge</i> <i>Infineon</i> BTS 7960 chip	27
2.11 Mikrokontroler Arduino Mega 2560.....	28
2.11.1 Daya	29
2.11.2 Memori.....	30
2.11.3 <i>Input & Output</i>	30
2.11.4 Komunikasi	31

2.12 Kontroler	31
2.13 Jenis-jenis Kontroler	33
2.13.1 Kontroler Proporsional (P).....	33
2.13.2 Kontroler Integral (I).....	34
2.13.3 Kontroler Differensial (D)	35
2.13.4 Kontroler Proporsional Integral Differensial	35
2.14 <i>Ziegler Nichols</i>	36
2.15 Transformasi Z	39
2.16 Diskritisasi	40
2.17 Parameter Performansi Sistem	41
BAB III METODE PENELITIAN	42
3.1 Perancangan Blok Diagram Sistem.....	43
3.2 Spesifikasi Desain	43
3.3 Karakteristik Setiap Blok	44
3.3.1 Karakterisasi Termokopel Tipe-K.....	44
3.3.2 Karakterisasi <i>Driver Motor</i>	47
3.3.3 Karakterisasi <i>Electric Fan</i>	49
3.4 Pembuatan Perangkat Keras.....	52
3.4.1 Desain Media	52
3.4.2 Prinsip Kerja Alat.....	53
3.4.3 Rangkaian Catu Daya	53
3.4.4 Konfigurasi Pin Arduino Mega 2560.....	54
3.4.5 Konfigurasi Pin <i>Driver Motor</i>	55
3.5 Perancangan Algoritma	56
3.5.1 Penentuan Karakteristik Pengontrolan Suhu.....	56
3.5.2 Parameter Kontroler PID dengan Metode <i>Ziegler-Nichols</i>	57
3.5.3 Diskritisasi	59
3.5.4 <i>Flowchat Program</i>	61
BAB IV HASIL DAN PEMBAHASAN	62
4.1 Pengujian Suhu Mesin Saat <i>Overhead</i>	62
4.2 Pengujian Penurunan Suhu Mesin Menggunakan Kontrol PID.....	63

4.3 Pengujian Penurunan Suhu Mesin Menggunakan Kontrol <i>On-Off</i> ...	64
4.4 Komparasi Kontrol PID dan Kontrol <i>On-Off</i>	65
BAB V PENUTUP.....	66
A. Kesimpulan.....	66
B. Saran	66
DAFTAR PUSTAKA	68
LAMPIRAN.....	69

DAFTAR TABEL

No	Judul	Halaman
Tabel 1.1	Hasil Pengujian Termokopel Tipe-K	12
Tabel 2.1	Fungsi pin pada <i>Interface Header High Current 30 A DC Motor Driver (datasheet)</i>	28
Tabel 2.2	Aturan Penalaan <i>Ziegler-Nichols</i> Berdasarkan Respon <i>Unit Step</i> dari <i>Plant</i>	10
Tabel 2.3	Aturan Dasar <i>Ziegler-Nichols</i> Berdasarkan <i>Critical Gain Kcr</i> dan <i>Critical</i>	11
Tabel 3.1	Hasil Pengujian Termokopel Tipe-K	12
Tabel 3.2	Hasil Pengujian <i>Driver Motor (High Current 30 A DC Motor Driver)</i>	48
Tabel 3.3	Hasil Pengujian <i>Electric Fan DC 12 V</i>	51
Tabel 3.4	Konfigurasi Pin Arduino Mega 2560 dalam Sistem	55
Tabel 3.5	Konfigurasi pin <i>Driver Motor H-Bridge BTS7960 30A</i> dalam Sistem	56
Tabel 3.6	Aturan Penalaan <i>Ziegler-Nichols</i> Berdasarkan Respon <i>Unit Step</i> Dari <i>Plant</i>	58
Tabel 4.1	Komparasi Kontrol PID dan Kontrol <i>On-Off</i>	65

DAFTAR GAMBAR

No	Judul	Halaman
Gambar 2.1	<i>Water Cooling System</i>	10
Gambar 2.2	Radiator	10
Gambar 2.3	Inti Radiator.....	11
Gambar 2.4	Tutup Radiator	12
Gambar 2.5	Pompa Air	13
Gambar 2.6	Kipas	14
Gambar 2.7	<i>Reservoir Tank</i>	15
Gambar 2.8	Mantel Pendingin	15
Gambar 2.9	<i>Thermostat</i>	16
Gambar 2.10	<i>Water Colling System</i>	19
Gambar 2.11	Sistem Pendingin Udara dengan <i>Electric Fan</i>	20
Gambar 2.12	Prinsip Kerja Termokopel	21
Gambar 2.13	Termokopel Tipe K	22
Gambar 2.14	<i>Max 6675 Modul</i>	24
Gambar 2.15	<i>Time Diagram Max 6675</i>	24
Gambar 2.16	<i>Motor Direct Current</i>	26
Gambar 2.17	<i>Electric Fan</i>	26
Gambar 2.18	Sinyal PWM	27
Gambar 2.19	<i>High Current 30A Motor Driver</i>	28
Gambar 2.20	<i>Arduino Mega 2560</i>	29
Gambar 2.21	<i>Diagram Blok Controller Proporsional</i>	33
Gambar 2.22	Diagram Blok Kontroler Proporsional	34
Gambar 2.23	Diagram Blok Kontroler Integral	34
Gambar 2.24	Diagram Blok Kontroler Differensial	35
Gambar 2.25	Diagram Blok Kontroler PID	36
Gambar 2.26	Kurva Respon <i>Unit Step</i> yang Menunjukkan 25% <i>Maximum Overshot</i>	36
Gambar 2.27	Respon <i>Plant</i> Terhadap Masukan Berupa <i>Unit Step</i>	37
Gambar 2.28	Kurva Respon yang Berbentuk S	37

Gambar 2.29	Sistem <i>Loop</i> Tertutup dengan Kontroler Proporsional	38
Gambar 2.30	Osilasi Berkesinambungan dengan PCR	39
Gambar 3.1	Blok Diagram Sistem Pendingin Mesin Mobil	43
Gambar 3.2	Blok Pengujian Termokopel Tipe K	45
Gambar 3.3	Grafik Hasil Pengujian Sensor Termokopel Tipe K	48
Gambar 3.4	Blok Pengujian <i>Driver Motor (High Current 30A</i> <i>DC Motor Driver)</i>	48
Gambar 3.5	Grafik Pengujian <i>Driver Motor (High Current 30A</i> <i>DC Motor Driver)</i>	49
Gambar 3.6	Blok Pengujian <i>Electric Fan</i> DC 12V	50
Gambar 3.7	Grafik Hasil Pengujian Kecepatan <i>Fan</i> Terhadap <i>Duty Cycle</i>	51
Gambar 3.8	Desain Alat	52
Gambar 3.9	Accu Motolite	54
Gambar 3.10	Skematik Konfigurasi I/O Arduino Mega 2560	54
Gambar 3.11	Respon Karakteristik Pengontrolan Suhu	56
Gambar 3.12	Metode <i>Ziegler-Nichols</i> Orde 1 pada Plant	57
Gambar 3.13	<i>Flowchat Program</i>	61
Gambar 4.1	Kondisi Suhu Mesin Saat <i>Overheat</i>	62
Gambar 4.2	Pengujian Plant dengan Kontrol PID	63
Gambar 4.3	Pengujian Plant dengan Kontrol <i>On-Off</i>	64

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Transportasi berasal dari kata Latin, yaitu *transportare*, di mana *trans* berarti seberang atau sebelah lain dan *portare* berarti mengangkut atau membawa. Transportasi berarti mengangkut atau membawa (sesuatu) ke sebelah lain atau suatu tempat ke tempat lainnya. Transportasi dapat didefinisikan sebagai usaha dan kegiatan mengangkut atau membawa barang dan/atau penumpang dari suatu tempat ke tempat lain (Abdul Kadir, 2006).

Pertumbuhan kendaraan bermotor di Indonesia tiap tahunnya mengalami kenaikan yang cukup signifikan. Pertumbuhan kendaraan bermotor ini didominasi oleh sepeda motor dan diikuti oleh kendaraan penumpang

Tabel 1.1 Jumlah Pertumbuhan Kendaraan di Indonesia (Badan Pusat Statistik)

Jenis Kendaraan	Jumlah Kendaraan (unit)			
	2011	2012	2013	2014
Mobil Penumpang	9548866	10432259	11484515	12599138
bis	2254406	2273821	2286309	2398846
Mobil Barang	4958738	5286061	5615494	6235136
Sepeda Motor	68839341	76381183	84732652	92976240
Jumlah	85601351	94373324	104118970	114209360

Kendaraan penumpang merupakan kendaraan yang sering digunakan dalam aktifitas sehari-hari. Dengan meningkatnya jumlah kendaraan bermotor setiap tahunnya maka kemacetan tidak dapat terhindarkan lagi, terutama bagi yang tinggal di kota besar, terutama pada jam sibuk. Beberapa penyebab kemacetan selain Karena volume kendaraan yang terus meningkat diantaranya adalah lebar jalan, kondisi jalan dan permukaan jalan.

Kemacetan membuat kendaraan menjadi sering *start-stop* dengan interval yang sangat sempit, hal ini dapat berdampak pada kendaraan bermotor. Salah satu dampak yang dapat terjadi adalah kenaikan suhu pada mesin. Semua mesin mobil pada saat ini telah memiliki sistem pendingin secara mekanik, diantaranya adalah *thermostat*, *radiator* dan kipas yang digerakkan oleh mesin. Sistem ini berguna untuk menjaga suhu optimal pada mesin yaitu berkisar antara 80°C-90°C (Agus Sudibyo, 2009). Akan tetapi tidak semua sistem pendingin tersebut mampu menjaga pada suhu optimal. Banyaknya gangguan seperti macet dan jalan menanjak konstan dapat memicu permasalahan ini. Maka dari itu diperlukan sistem pendingin tambahan yang membantu kerja sistem pendingin yang telah ada.

Oleh Karena itu, untuk memecahkan masalah diatas diantaranya adalah dengan memasang *electric fan* yang akan membantu menjaga agar suhu tidak meningkat sehingga suhu mesin tetap pada suhu optimal, dibuatlah skripsi ini dengan judul pengaman suhu mesin mobil secara otomatis dengan menggunakan kontroler PID. Kontroler merupakan suatu sistem yang dinamis yang sengaja ditambahkan untuk mendapatkan karakteristik sistem keseluruhan yang diinginkan (Ogata K, 2010).

Mobil dipilih untuk penelitian ini dikarenakan mobil adalah alat transportasi yang umum di berbagai negara, jumlah mobil terus meningkat setiap tahunnya dan sistem ini dapat diaplikasikan pada semua mobil.

Alat yang digunakan untuk sistem pengaman suhu mobil otomatis ini menggunakan kontroler PID berbasis Arduino Mega 2560. Proporsional Integral Differensial (PID) adalah kontroler yang merupakan gabungan dari kontroler proporsional, kontroler integral, dan kontroler differensial. Gabungan dari ketiga kontroler ini diharapkan dapat memberikan keluaran sistem yang stabil dimana masing-masing kontroler bisa saling melengkapi satu dengan yang lainnya. Keuntungan dari kontroler PID adalah sebuah sistem yang sederhana sehingga dapat lebih cepat untuk mencapai keluaran sistem dengan *error* yang kecil. Diharapkan dengan menggunakan kontroler PID suhu mesin mobil dapat terjaga

dan terhindarkan dari kerusakan mesin mobil yang membutuhkan biaya besar dan waktu yang lama.

1.2 Rumusan Masalah

Mengacu pada permasalahan yang telah diuraikan pada latar belakang, maka rumusan masalah dapat ditekankan pada point berikut:

1. Bagaimana perancangan sistem dan pembuatan alat pendinginan suhu mesin kendaraan bermotor roda 4 ?
2. Bagaimana merancang software untuk pengontrolan suhu dan mendapatkan nilai-nilai parameter Kontroler PID agar sesuai dengan yang diinginkan ?
3. Bagaimana Kelebihan Kontroler *PID* dibanding dengan Kontrol *On-Off* ?

1.3 Batasan Masalah

Mengacu pada permasalahan pada proposal ini, maka akan dibatasi pada:

1. Mikrokontroller yang digunakan adalah Arduino Mega 2560
2. Sensor yang digunakan suhu yang digunakan adalah yang terdapat pada mobil
3. Mobil yang digunakan adalah Suzuki 1300cc Bensin Injection dengan kode mesin G13A
4. Pembahasan ditekankan pada penggunaan kontroler PID untuk efisiensi
5. *Electric Fan* yang digunakan mempunyai spesifikasi 12v – 80 W – 6A, diameter 27cm ketebalan 7cm
6. Kinerja rangkaian driver dan elektronika tidak dibahas mendalam

1.4 Tujuan Penelitian

Tujuan pemecahan masalah penelitian ini adalah meminimalisir kerusakan mesin mobil karena kenaikan suhu dan bertambahnya masa pakai mobil.

1.5 Manfaat Penelitian

Manfaat penelitian ini adalah dapat dipahaminya pengontrolan suhu pada mesin mobil dengan menggunakan *Electric Fan* untuk menjaga suhu mobil tetap pada suhu optimal dengan menggunakan metode Kontroler Proporsional Integral Differensial (PID)

BAB II

TINJAUAN PUSTAKA

Bab ini menjelaskan teori-teori yang mendukung skripsi, yang terdiri atas:

2.1 Transportasi

Transportasi merupakan hal yang penting dalam berbagai aspek Transportasi sensiri berasal dari kata Latin, yaitu *transportare*, di mana *trans* berarti seberang atau sebelah lain dan *portare* berarti mengangkut atau membawa. Transportasi berarti mengangkut atau membawa (sesuatu) ke sebelah lain atau suatu tempat ke tempat lainnya. Transportasi dapat didefinisikan sebagai usaha dan kegiatan mengangkut atau membawa barang dan/atau penumpang dari suatu tempat ke tempat lain (Abdul Kadir, 2006). Pada era saat ini transportasi merupakan sarana yang sangat penting dalam perekonomian suatu negara.

Fungsi utama transportasi ada dua, yaitu sebagai penunjang (*servicing facility*) dan sebagai pendorong atau pendukung (*promoting facility*) (M.N. Nasution, 1996). Pertama, transportasi berfungsi sebagai penunjang (*servicing facility*) dimaksudkan jasa transportasi itu melayani pengembangan kegiatan sektor-sektor lain yaitu sektor-sektor pertanian, industry perdagangan pendidikan, kesehatan, pariwisata, transmigrasi dan lainnya. Kedua, transportasi berfungsi sebagai pendorong pembangunan (*promoting facility*), dimaksudkan bahwa pengadaan/pembangunan fasilitas (prasarana dan sarana) transportasi diharapkan dapat membantu membuka keterisolasian, keterpencilan, keterbelakangan daerah-daerah serta daerah-daerah perbatasan. Daerah terisolasi adalah daerah yang belum terjangkau oleh pelayanan transportasi, karena belum tersedia fasilitas transportasi, meskipun letaknya tidak jauh dari pusat kegiatan/pelayanan.

Tanpa adanya transportasi sebagai sarana pendukung tidak dapat diharapkan tercapainya hasil yang memuaskan dalam usaha pengembangan ekonomi dari suatu negara.

Transportasi dapat diklasifikasikan menurut sudut jalan atau permukaan jalan yang dipakai dan tenaga penggerak yang digunakan, yaitu:

1. Transportasi jalan raya

Transportasi melalui darat atau *land transportation* ini terdiri atas:

- a. Transportasi jalan raya

Dalam transportasi jalan raya (*road transport*) meliputi transportasi menggunakan alat angkut yang berupa manusia, kendaraan bermotor dan hewan.

- b. Transportasi jalan rel

Dalam transportasi jalan rel (*rail transport*) ini digunakan angkutan berupa kereta api, yang terdiri dari lokomotif, gerbong, dan kereta penumpang.

2. Transportasi melalui air

Transportasi melalui air (*water transport*) terdiri atas dua macam, yaitu:

- a. Transport air pedalaman

Transport ini melalui air pedalaman (*inland transport*) menggunakan alat angkutan berupa sampan, kano, perahu motor dan kapal.

- b. Transport laut

Di dalam transport laut (*ocean transport*) digunakan alat angkutan perahu, kapal api/uap, dan kapal mesin.

2.1.1 Kendaraan Bermotor Roda 4

Dari penjelasan tentang transportasi diatas penelitian ini berfokus pada pengguna kendaraan di jalan raya, khususnya adalah mobil. Kendaraan bermotor adalah kendaraan yang digerakkan oleh peralatan teknik yang berada pada kendaraan itu (Peraturan Pemerintah Republik Indonesia Nomor 44 Tahun 1993). Kendaraan bermotor dikelompokkan dalam beberapa jenis, yaitu:

- a. Sepeda Motor
- b. Mobil Penumpang
- c. Mobil Bus
- d. Mobil Barang
- e. Kendaraan Khusus

Mobil penumpang merupakan alat transportasi yang sering digunakan di berbagai negara. Kendaraan ini sering digunakan untuk aktifitas sehari-hari, bepergian dan berbagai aktifitas lainnya. Dengan meningkatnya jumlah kendaraan dari berbagai jenis maka hal itu dapat menyebabkan kemacetan. Kemacetan sendiri memiliki berbagai dampak pada tiap kendaraan bermotor.

Dalam penelitian ini dipilih kendaraan bermotor berjenis mobil penumpang dikarenakan kendaraan tersebut merupakan alat transportasi manusia yang umum digunakan.

2.2 Mesin Kendaraan Bermotor Roda 4

Perkakas untuk menggerakkan atau membuat sesuatu yang dijalankan dengan roda, digerakkan oleh tenaga manusia atau motor penggerak, menggunakan bahan bakar minyak atau tenaga alam (Kamus Besar Bahasa Indonesia).

Mesin kendaraan bermotor baik diesel maupun bensin dikategorikan sebagai mesin kalor. Yang dimaksud mesin kalor disini adalah mesin yang menggunakan energi termal untuk menghasilkan kerja mekanik. Motor Selanjutnya jika ditinjau dari cara memperoleh sumber energi termal, jenis mesin kalor dapat dibagi menjadi dua bagian, yaitu Motor Pembakaran Luar (*External Combustion Engine*) dan Motor Pembakaran Dalam (*Internal Combustion Engine*). Mesin pembakaran ini pada umumnya dikenal dengan motor bakar.

Motor bakar adalah suatu bagian dari mesin kalor yang berfungsi untuk mengkonversi energi termal hasil pembakaran bahan bakar menjadi energi mekanis. Terjadinya energi panas karena adanya proses pembakaran, bahan bakar, udara, dan sistem pengapian. Dengan adanya suatu konstruksi mesin, memungkinkan terjadinya siklus kerja mesin untuk usaha dan tenaga dorong dari hasil ledakan pembakaran yang diubah oleh konstruksi mesin menjadi energi mekanik atau tenaga penggerak. Berikut adalah jenis dari motor bakar:”

A. Motor Pembakaran Luar (*External Combustion Engine*)

Motor pembakaran luar adalah suatu motor dimana proses pembakaran atau perubahan energi panas dilakukan di luar dari mekanisme/konstruksi mesin. Dari ruang pembakaran energi panas tersebut dialirkan ke konstruksi mesin melalui media penghubung lagi. Contoh motor pembakaran luar adalah mesin uap/turbin uap dan Mesin Nuklir/Turbin Nuklir.

B. Motor Pembakaran Dalam (*Internal Combustion Engine*)

Pada motor pembakaran dalam, proses pembakaran atau perubahan energi panas dilakukan di dalam konstruksi mesin itu sendiri dan tempat terjadinya proses pembakaran itu disebut ruang bakar. Contohnya adalah motor bensin, motor diesel, dan mesin Jet.

Motor bakar pada umumnya dibedakan menjadi dua, yaitu Motor Bensin (Otto) dan Motor Diesel. Perbedaan kedua jenis motor tersebut sangat jelas sekali yaitu jika motor bensin menggunakan bahan bakar bensin (premium), sedangkan motor diesel menggunakan bahan bakar solar. Perbedaan yang utama juga terletak pada sistem penyalanya, di mana pada motor bensin digunakan busi sebagai sistem penyalanya sedangkan pada motor diesel memanfaatkan suhu kompresi yang tinggi untuk dapat membakar bahan bakar solar.

a. Motor Bensin

Yang menjadi ciri utama dari motor bensin adalah proses pembakaran bahan bakar yang terjadi di dalam ruang silinder pada volume tetap. Proses pembakaran pada volume tetap ini disebabkan pada waktu terjadi kompresi, dimana campuran bahan bakar dan udara mengalami proses kompresi di dalam silinder, dengan adanya tekanan ini bahan bakar dan udara dalam keadaan siap terbakar dan busi meloncatkan bunga listrik sehingga terjadi pembakaran dalam waktu yang singkat sehingga campuran tersebut terbakar habis seketika dan menimbulkan kenaikan suhu dalam ruang bakar.

b. Motor Diesel

Motor diesel memiliki ciri utama yaitu pembakaran bahan bakar di dalam silinder berlangsung pada tekanan konstan, dimana gas yang dihisap pada langkah hisap yang merupakan udara murni tersebut berada di dalam silinder pada waktu piston berada di titik mati atas. Bahan bakar yang masuk kedalam silinder oleh *injector* terbakar bersama dengan udara oleh suhu kompresi yang tinggi.

2.3 Sistem Pendingin Mobil

Setiap kendaraan bermotor roda 4 telah memiliki sistem pendingin produksi pabrik. Sistem pendingin ini bertujuan agar mesin selalu pada suhu optimal. Apabila suhu terlalu tinggi maka dapat mengakibatkan kerusakan pada mobil, dan apabila dibawah suhu kerja optimal maka juga memiliki dampak. Jika mesin bekerja pada suhu yang rendah akan membuat komponen mesin cepat mengalami kerusakan, detonasi, membuat polusi dan boros bahan bakar. (Sudibyo, 2009)

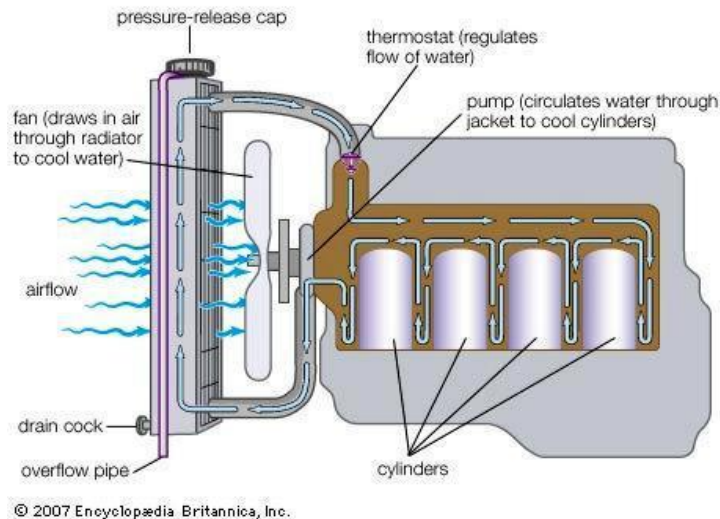
Sistem pendingin mobil pada umumnya memiliki 2 tipe pendingin, yaitu:

A. Pendingin Air (*Water Cooling System*)

Panas dilewatkan atau ditransfer ke air disekitar ruang bakar dan silinder. Panas yang diserap oleh air pendingin akan menyebabkan naiknya temperatur air pendingin tersebut. Panas dari air ditransfer ke sirip radiator kemudian panas tersebut disemburkan ke udara, air kemudian kembali ke mesin (Daryanto, 2002).

Sistem pendinginan air mempunyai keuntungan antara lain lebih aman, karena ruang bakar dikelilingi oleh pendingin (terutama air dengan *adiktive* dan anti beku). Sistem pendinginan air disamping lebih aman juga mempunyai keuntungan lain seperti sebagai peredam bunyi, air dingin yang panas dapat digunakan sebagai sumber panas untuk pemanas udara di dalam kendaraan. Pengontrolan suhu pendinginan dalam sistem ini lebih mudah dibandingkan dengan sistem pendinginan udara karena pada sistem pendinginan terdapat *thermostat*, pendinginan lebih merata, suhu kerja lebih cepat tercapai karena adanya *thermostat* yang akan bekerja pada waktu suhu mesin rendah.

Sistem pendinginan air juga mempunyai kerugian antara lain lebih rumit dan lebih mahal dibandingkan dengan sistem pendingin udara (Daryanto, 2002).



Gambar 2.1 Water Cooling System
Sumber: *Encycloprdia Britania, Inc.* (2007)

Komponen Sistem Pendinginan Air

1. Radiator

Radiator berfungsi sebagai alat untuk mendinginkan air pendingin yang menyerap panas dari mesin dengan cara membuang panas tersebut melalui sirip-sirip radiator. Air dari radiator tersebut dikirim ke bagian yang didinginkan melalui selang radiator, baik dari radiator ke blok silinder ataupun dari blok silinder ke radiator.



Gambar 2.2 Radiator
Sumber: Irfan, S Ade. (2007)

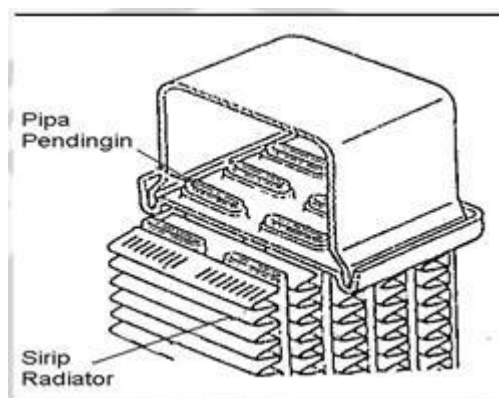
Konstruksi radiator terdiri dari:

a. Tangki Atas

Tangki atas berfungsi untuk menampung air panas dari mesin. Tangki ini juga dilengkapi dengan lubang pengisian, pipa pembuangan dan saluran masuk air dari mesin. Pipa pembuangan berhubungan dengan tangki *reservoir* untuk membuang kelebihan air sehingga tidak terdapat gelembung air dalam sistem.

b. Inti Radiator

Inti radiator berfungsi untuk membuang panas dari air ke udara agar temperatur menjadi lebih rendah dari sebelumnya. Inti radiator terdiri dari pipa-pipa air untuk mengalirkan air dari tangki atas ke tangki bawah dan sirip-sirip pendingin untuk membuang panas air yang ada pada pipa.



Gambar 2.3 Inti radiator
Sumber: Irfan, S Ade. (2007)

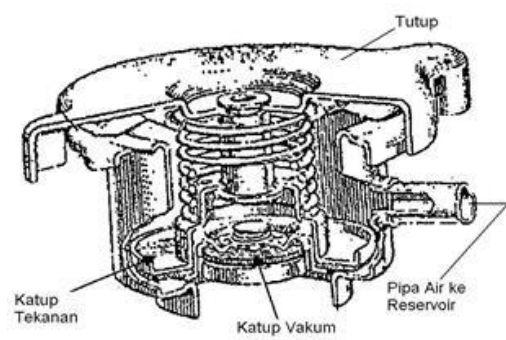
c. Tangki Bawah

Tangki bawah berfungsi untuk menampung air yang telah didinginkan oleh inti radiator dan selanjutnya disalurkan ke mesin melalui kerja pompa. Selain itu tangki bawah juga berhubungan dengan saluran pembuangan air pada saat dilakukan pengurasan air radiator.

2. Tutup Radiator

Fungsi tutup radiator antara lain:

- Penutup radiator agar tidak terjadi kebocoran.
- Membuat sistem menjadi bertekanan sehingga dapat mencegah terjadinya penguapan air dalam sistem (fungsi *relief valve*) dan mempercepat pencapaian suhu kerja mesin.
- Untuk mengurangi tekanan apabila tekanan di dalam sistem berlebihan sehingga dapat mencegah kerusakan dari bagian sistem.
- Mengalirkan air dari radiator ke penampung atau *reservoir* dan memasukkan kembali pada saat tekanan dalam radiator turun (fungsi katub *vacum*).

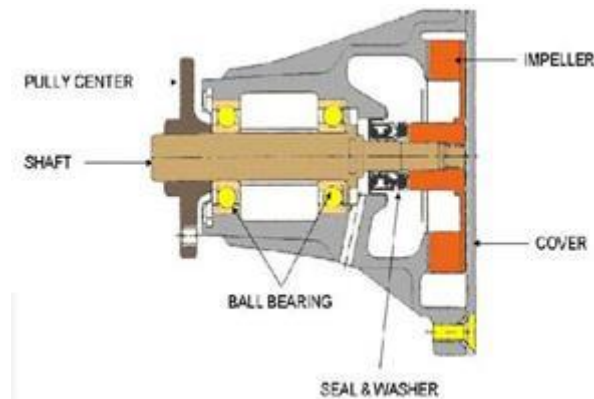


Gambar 2.4. Tutup Radiator
Sumber: Irfan, S Ade. (2007)

3. Pompa air

Pompa air berfungsi untuk mensirkulasikan air pendingin dari radiator ke silinder mesin. Pompa air yang digunakan dalam sistem pendingin Isuzu Panther adalah pompa tipe sentrifugal yang akan dipasangkan pada bagian depan blok silinder. Gerak putar pompa diperoleh dari putaran poros engkol melalui *pully* yang dihubungkan dengan *belt*.

Kelebihan dari pompa air mesin Isuzu Panther adalah mempunyai tekanan yang tinggi. Kerugiannya apabila terjadi kerusakan maka harus dilakukan penggantian satu unit pompa, sehingga jenis pompa ini tidak boleh dilakukan pembongkaran. Dalam pemasangannya pada kepala silinder, pompa ini dilengkapi dengan gasket yang berguna untuk mencegah terjadinya kebocoran air pendingin.



Gambar 2.5 Pompa air
Sumber: Irfan, S Ade. (2007)

Bagian-bagian pompa dari air :

a. Poros (*shaft*)

Merupakan komponen utama pada pompa dimana bagian depannya dihubungkan dengan puli untuk mendapatkan tenaga dari putaran poros engkol sedangkan bagian belakang dihubungkan dengan *impeler* pompa.

b. *Impeler*

Impeler berfungsi untuk membuat perbedaan tekanan pada saat pompa bekerja.

c. *Water Pump seal*

Water pump seal berfungsi untuk mencegah kebocoran air dari sistem pendingin pada poros pompa.

4. Kipas Pendingin

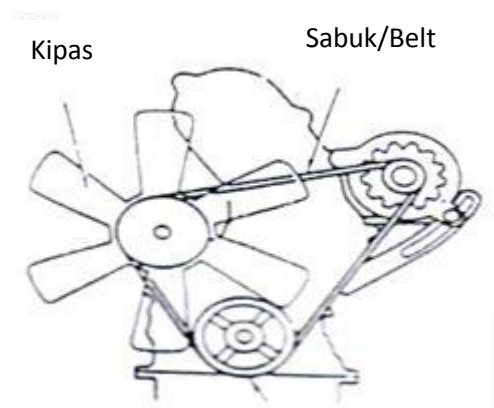
Untuk memastikan aliran udara yang benar melalui inti radiator dan sekitar mesin, pasang kipas dengan engkol mesin dan roda-roda puli. Kipas ini terdiri dari sebuah daun atau bilah yang terbuat dari baja tipis atau bahan plastik. Ketika baling-baling berputar, bilah (*blade*) menggerakkan udara ke dalam unit mesin.

Kipas mempunyai tiga hal yang tidak menguntungkan yaitu:

- Berisik
- Menyerap tenaga mesin sebesar 2-3 tenaga kuda (tk)
- Dalam keadaan panas atau dingin pendinginan tetap diperlukan, ketika mesin dinyalakan dalam kondisi dingin, kipas angin dengan segera menyebarkan udara dan menambah waktu pemanasan.

Bilah kipas yang fleksibel terbuat dari plastik sehingga ketika kecepatan bertambah apabila sudut bilah digerakan oleh tekanan udara. Metode ini mempunyai efek sebagai berikut:

- Operasinya tenang pada saat mesin berkecepatan lebih tinggi
- Sedikit tenaga yang diserap dengan kecepatan tinggi, bilah yang rata menggantikan sedikit udara dan untuk itu sedikit usaha diperlukan untuk memutar kipas.
- Bilah plastik berbentuk lempeng, untuk mengurangi aliran udara di atas mesin dan juga beban ditempatkan pada mesin oleh kipas (Daryanto, 2002).

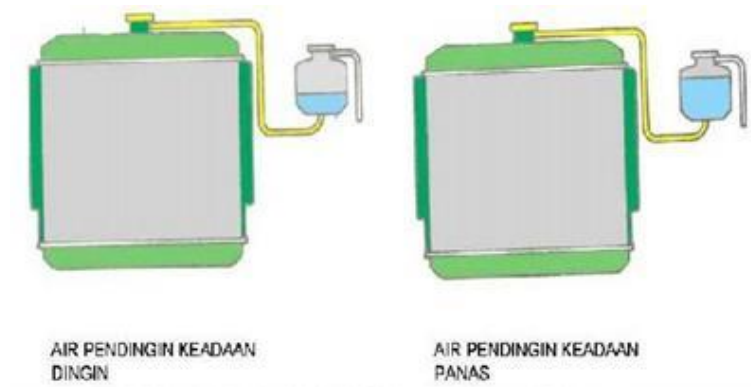


Gambar 2.6 Kipas
Sumber: Irfan, S Ade. (2007)

5. Tangki Cadangan

Tangki cadangan (*reservoir tank*) dihubungkan ke radiator dengan selang *over flow*. Apabila tempertur dan tekanan air pendingin naik menyebabkan cairan pendingin berekspansi. Saat tekanan dan volume melebihi kemampuan kerja tutup radiator maka cairan pendingin yang berlebihan akan dikirim ke *reservoir*. Apabila temperatur turun, maka cairan

pendingin yang ada di dalam tangki cadangan akan kembali ke radiator. Hal ini untuk mencegah terbuangnya cairan pendingin saat diperlukan agar jumlahnya tetap.

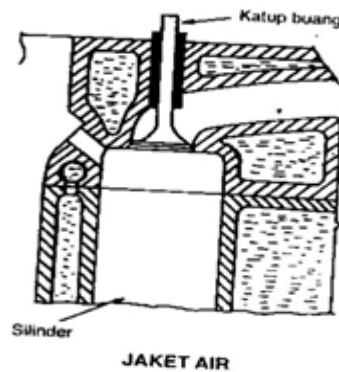


Gambar 2.7 *Reservoir Tank*

Sumber: Irfan, S Ade. (2007)

6. Mantel Pendingin

Mantel pendingin pada mesin Isuzu Panther mengelilingi silinder-silinder dan kepala silinder, yang berfungsi untuk mendinginkan bagian-bagian pendingin silinder dan ruang bakar secara efektif. Mantel pendingin pada kepala silinder dan blok silinder berhubungan langsung dengan tangki radiator bagian atas



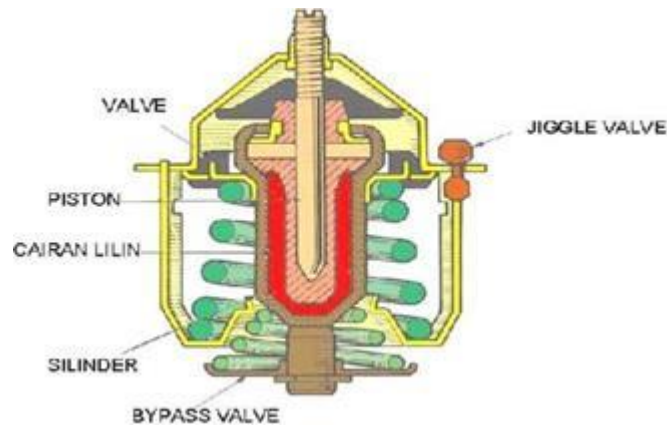
Gambar 2.8 Mantel Pendingin

Sumber: Irfan, S Ade. (2007)

7. Thermostat

Thermostat dipasang pada blok silinder bagian atas atau lebih tepatnya pada *water inlet housing* dengan sambungan selang yang menyalurkan air dari mesin ke radiator. *Thermostat* tidak bekerja pada suhu di bawah 80°C . Pada suhu tersebut *Thermostat* menutup

sehingga air hanya beredar di sekeliling blok silinder tidak sampai ke radiator, dengan demikian suhu motor dapat dikendalikan.



Gambar 2.9 Thermostat.

Sumber: Irfan, S Ade. (2007)

Fungsi *Thermostat* yaitu untuk mengendalikan suhu mesin hingga mencapai suhu kerja. Jenis *Thermostat* yang digunakan pada mesin Isuzu Panther adalah tipe *wax pellet*. Tipe *wax pellet* ini menggunakan semacam lilin yang dapat mengembang pada saat dipanaskan dan akan menyusut pada waktu dingin.

Cara kerja:

Pada saat air pendingin panas kurang lebih pada suhu 80° C, *Wax Pellet* yang ada dalam *thermostat* akan memuai dan mendorong katup untuk membuka.

Hal ini disebabkan karena pemuai lilin tersebut mampu menahan pegas, *thermostat*. Pada saat temperatur air pendingin telah dingin, maka *Wax Pellet* di dalam *thermostat* akan menyusut, sehingga pegas akan mendorong katup *thermostat* untuk menutup kembali.

8. Pipa-Pipa Saluran (Selang)

Pemasangan saluran pendingin memerlukan pipa saluran yang fleksibel, seperti saluran utama bagian atas dan bagian bawah radiator serta saluran *bypass* dan saluran lainnya bisa digunakan untuk memindahkan zat pendingin menuju atau keluar dari mesin.

Saluran radiator membentuk suatu hubungan fleksibel dengan mesin dan radiator, sehingga memungkinkan untuk disirkulasikan dan meredam dari getaran mesin yang bergerak. Pipa atau selang terbuat dari karet, agar dapat menjaga kestabilan temperatur, dan tekanan dalam sistem (Daryanto, 2002).

Bagian luar selang dibalut dengan selang penjepit yang berfungsi: membalut permukaan, menjaga tekanan dalam sistem dengan menahan kelenturannya dan menjadi peredam suhu dalam sistem pendinginan (Daryanto, 2002).

Macam-macam selang dalam sistem pendingin antara lain :

a. Selang Radiator atas

Selang radiator atas berfungsi menghubungkan bagian atas dari radiator ke pengeluar (*outlet*) ruang pengukur panas dan menyalurkan air panas dari mesin ke radiator.

b. Selang radiator bawah

Selang radiator bawah berfungsi menghubungkan bagian lebih rendah pada ruang *thermostat* ke sisi jalan masuk pompa air dan menyalurkan air hangat dari radiator ke mesin.

c. Selang *bypass* (ketika dipasang)

Selang *bypass* (ketika dipasang) berfungsi untuk menghubungkan bagian lebih rendah pada ruang *thermostat* ke sisi jalan masuk pompa air dan menyediakan sirkulasi ke pompa ketika *thermostat* tertutup.

d. Selang pemanas

Selang pemanas biasanya digunakan untuk mengedarkan air ke pemanas kendaraan atau saluran masuk pompa. Satu selang menghubungkan bagian terendah ruang *thermostat* atau kepala silinder dan melangsungkan air panas ke pemanas. Selang yang lain menghubungkan ke sisi jalan masuk pada pompa air untuk menyalurkan air hangat kembali ke mesin.

e. Selang penjepit

Selang penjepit digunakan untuk melindungi kerapatan selang untuk macam-macam hubungan (pada ujung selang). Beberapa jenis dari selang penjepit pada kendaraan antara lain jubilee, tipe skrup, dan tipe kancing atau spring.

9. Coolant

Coolant ini mempunyai kualitas tinggi berbahan dasar *ethilene glycol non-silicate, non amine, non nitrite, non borate* dengan *long-life hybrid organic acid technology*. Keuntungan pendingin dengan menggunakan *coolant* yaitu :

- a. Air pendingin tidak dapat membeku.
- b. *Coolant* bebas kapur, sehingga saluran-saluran selalu bersih.
- c. Mencegah terjadi korosi pada komponen-komponen sistem pendingin.
- d. Melumasi *thermostat* dan pompa air sehingga tidak macet.
- e. Mencegah *over heating* pada mesin.

B. Pendingin udara (*Air Cooling System*)

Panas dari mesin yang bekerja atau berputar, dilewatkan melalui sirip, rusuk, atau *fins* ke udara luar.

Dasar penggunaan pada sistem pendinginan udara ini tergantung pada hal sebagai berikut

- a. Perbedaan temperatur antara panas mesin dengan udara luar/sekitar
- b. Luas permukaan dimana panas dikeluarkan atau disebarkan
- c. Tingkat aliran udara pada permukaan yang dikenai (Daryanto, 2002).

Penggunaan sistem pendinginan udara mempunyai keuntungan seperti bahan bakar hemat dan keausan silinder berkurang, tidak ada bahaya kerusakan karena pembekuan.

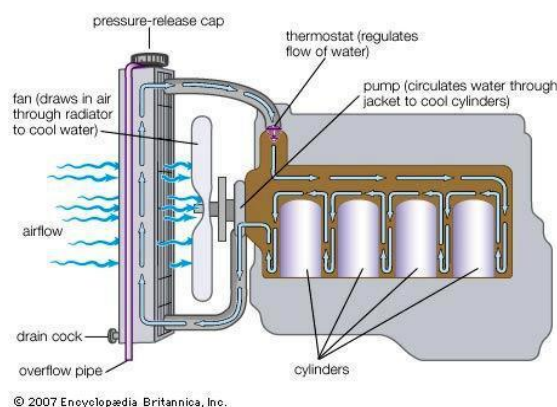
2.4 Gabungan Pendingin Air dan Pendingin Udara

Dalam penelitian ini digunakan gabungan antara sistem pendingin air dan sistem pendingin udara. Kedua sistem ini akan bekerja secara bersamaan untuk menjaga agar sistem suhu pada mesin tetap pada suhu optimal.

Dalam penelitian ini *Electric fan* akan berperan sebagai sistem pendingin udara tambahan. *Electric Fan* ini digerakkan oleh motor DC 12V yang dicatu oleh aki mobil. Penempatan alat ini berada pada bagian depan radiator mobil, pada saat menyala maka akan menghembuskan udara ke sirip-sirip radiator yang akan membantu proses pendinginan. akan menyala pada saat suhu mencapai 87°C dan akan pada apabila suhu mencapai 86°C

Untuk mencapai hasil yang maksimal dalam percobaan ini maka setiap komponen harus diuji. Sistem ini tidak akan bekerja secara sempurna apabila ada komponen yang tidak berfungsi. Apabila hanya satu sistem saja yang berfungsi maka pendinginan tidak akan bekerja secara maksimal.

Dengan gabungan kedua sistem ini diharapkan suhu mesin selalu pada suhu optimal.

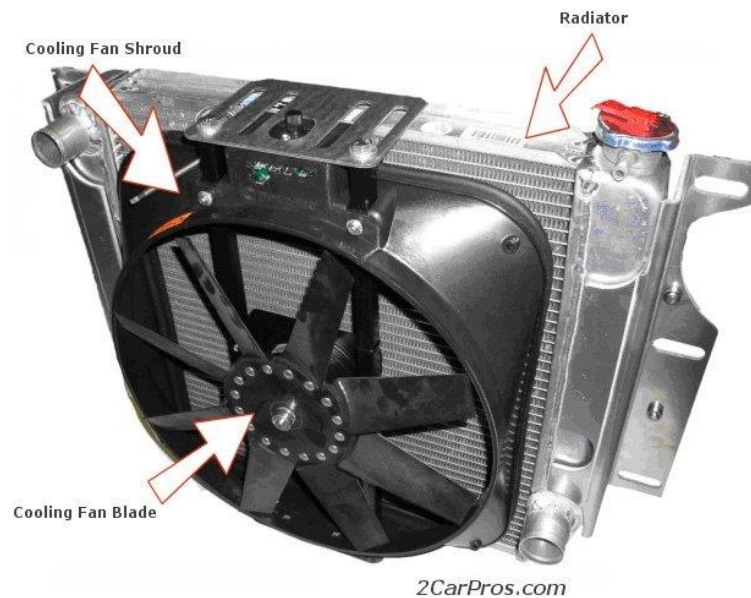


Gambar 2.10 Water Cooling System

Sumber: *Encyclopedia Britania, Inc*

Pada gambar 2.10 menunjukkan bahwa airflow dari depan radiator hanya didapatkan dari lingkungan sekitar. Dengan pemasangan *Electric Fan* pada bagian depan radiator maka airflow dari arah depan menuju radiator akan selalu tersedia saat dibutuhkan. Dengan gabungan sistem pendingin yang bekerja beriringan ini diharapkan akan menjaga suhu kendaraan pada suhu optimal.

Berikut pada gambar 2.11 menunjukkan pemasangan *Electric Fan* pada bagian depan radiator



Gambar 2.11 Sistem pendingin udara dengan *electric fan*

Sumber: www.2carpros.com

2.5 Sensor Termokopel (*Thermocouple Sensor*)

Sensor termokopel (*Thermocouple Sensor*) adalah jenis sensor suhu yang digunakan untuk mendeteksi atau mengukur suhu melalui dua jenis logam konduktor berbeda yang digabung pada ujungnya sehingga menimbulkan efek "*Thermo-electric*". Efek *Thermo-electric* pada Termokopel ini ditemukan oleh seorang fisikawan Estonia bernama *Thomas Johann Seebeck* pada Tahun 1821, dimana sebuah logam konduktor yang diberi perbedaan panas secara gradient akan menghasilkan tegangan listrik. Perbedaan Tegangan listrik diantara dua persimpangan (junction) ini dinamakan dengan Efek "*Seebeck*".

Termokopel merupakan salah satu jenis sensor suhu yang paling populer dan sering digunakan dalam berbagai rangkaian ataupun peralatan listrik dan Elektronika yang berkaitan dengan Suhu (Temperature). Beberapa kelebihan Termokopel yang membuatnya menjadi populer adalah responnya yang cepat terhadap perubahan suhu dan juga rentang suhu operasionalnya yang luas yaitu berkisar diantara -200°C hingga 2000°C . Selain respon

yang cepat dan rentang suhu yang luas, Termokopel juga tahan terhadap guncangan/getaran dan mudah digunakan.

2.5.1 Prinsip Kerja Termokopel

Prinsip kerja Termokopel cukup mudah dan sederhana. Pada dasarnya Termokopel hanya terdiri dari dua kawat logam konduktor yang berbeda jenis dan digabungkan ujungnya. Satu jenis logam konduktor yang terdapat pada Termokopel akan berfungsi sebagai referensi dengan suhu konstan (tetap) sedangkan yang satunya lagi sebagai logam konduktor yang mendeteksi suhu panas.

Untuk lebih jelas mengenai Prinsip Kerja Termokopel, mari kita melihat gambar dibawah ini :

Termokopel (Thermocouple)



Gambar 2.12 Prinsip Kerja Termokopel

Sumber: *datasheet Thermocouple type-K*

Berdasarkan Gambar diatas, ketika kedua persimpangan atau Junction memiliki suhu yang sama, maka beda potensial atau tegangan listrik yang melalui dua persimpangan tersebut adalah “NOL” atau $V_1 = V_2$. Akan tetapi, ketika persimpangan yang terhubung dalam rangkaian diberikan suhu panas atau dihubungkan ke obyek pengukuran, maka akan terjadi perbedaan suhu diantara dua persimpangan tersebut yang kemudian menghasilkan tegangan listrik yang nilainya sebanding dengan suhu panas yang diterimanya atau $V_1 - V_2$. Tegangan Listrik yang ditimbulkan ini pada umumnya sekitar $1\ \mu\text{V} - 70\ \mu\text{V}$ pada tiap derajat Celcius. Tegangan tersebut kemudian dikonversikan sesuai dengan Tabel referensi yang telah ditetapkan sehingga menghasilkan pengukuran yang dapat dimengerti.

2.5.2 Termokopel Tipe K

Termokopel tersedia dalam berbagai ragam rentang suhu dan jenis bahan. Pada dasarnya, gabungan jenis-jenis logam konduktor yang berbeda akan menghasilkan rentang suhu operasional yang berbeda pula, salah satunya adalah termokopel tipe K. Termokopel tipe K terdiri dari; nikel dan kromium pada sisi positif (Thermocouple Grade) sedangkan sisi negatif negatif (Extension Grade) terdiri dari nikel dan alumunium. Thermocouple jenis ini sering dipakai pada tujuan umum dikarenakan cenderung lebih murah. Tersedia untuk rentang suhu 0 °C hingga +1100 °C. Berikut adalah gambar Termokopel tipe K :



Gambar 2.13 Termokopel Tipe-K

Spesifikasi alat yang digunakan pada skripsi ini sebagai berikut :

Range 0°C – 400°C

Probe berbentuk fitted bolts

Berbahan Chrome Alumel

2 konduktor sebagai output kabel

2.6 Max 6675

MAX 6675 dibentuk dari kompensasi cold-junction yang outputnya didigitalisasi dari sinyal termokopel tipe-K. data output memiliki resolusi 12-bit dan mendukung komunikasi SPI mikrokontroler secara umum. Data dapat dibaca dengan mengkonversi hasil pembacaan 12-bit data. MAX6675 memiliki bagian ujung *cold end* yang hanya dapat mengukur -20°C sampai $+85^{\circ}\text{C}$. Pada saat bagian *cold end* MAX6675 mengalami fluktuasi suhu maka MAX6675 akan tetap dapat mengukur secara akurat perbedaan temperature pada bagian yang lain. MAX6675 dapat melakukan koreksi atas perubahan pada temperature ambient dengan kompensasi *cold-junction*. Device mengkonversi temperature ambient yang terjadi ke bentuk tegangan menggunakan sensor temperature diode. Untuk dapat melakukan pengukuran actual, MAX6675 mengukur tegangan dari output termokopel dan tegangan dari sensing diode.

Performance optimal MAX6675 dapat tercapai pada waktu termokopel bagian *cold-junction* dan MAX6675 memiliki temperature yang sama. Hal ini untuk menghindari penempatan komponen lain yang menghasilkan panas didekat MAX6675.

Penguatan Sinyal

Untuk mendapatkan nilai suhu pada termokopel tipe-K, perubahan tegangan sebesar $41\mu\text{V}/^{\circ}\text{C}$ dengan menggunakan pendekatan karakteristik dapat menggunakan rumus dibawah ini:

$$V_{\text{OUT}} = (41\mu\text{V} / ^{\circ}\text{C}) 5 (T_{\text{R}} - T_{\text{AMB}})$$

Dengan:

$$V_{\text{OUT}} = \text{Voutput termokopel } (\mu\text{V})$$

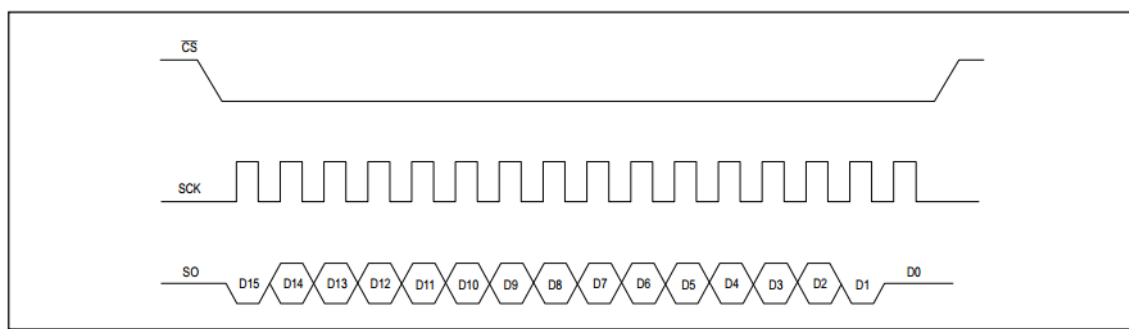
T_{R} adalah termperatur remote junction ($^{\circ}\text{C}$)

T_{AMB} adalah temperature ambient ($^{\circ}\text{C}$)



Gambar 2.14 Max6675 Modul

Sumber: *datasheet max6675*



Gambar 2.15 Timing Diagram Max6675

Sumber : *datasheet max6675*

2.7 Mesin *Direct Current* (DC)

Mesin *Direct Current* (DC) merupakan alat atau perangkat yang memiliki fungsi untuk mengkonversikan energi mekanik menjadi energi listrik begitu pula sebaliknya. Mesin DC memiliki 2 elemen yakni stator dan rotor. Stator merupakan elemen mesin DC yang berdiri tetap/statis, sedangkan rotor merupakan elemen mesin DC yang berputar/dinamis. Stator sendiri terdiri dari beberapa bagian yaitu gandar, kutub utama, kutub komutasi/kutub bantu, dan kumparan penguat. Sementara itu, rotor terdiri dari inti jangkar, kumparan jangkar, kawat baja pengikat, dan komutator. Selain itu, mesin DC juga terdiri dari celah udara dan sikat-sikat.

2.7.1 Motor *Direct Current* (DC)

Mesin *Direct Current* (DC) yang memiliki fungsi untuk mengkonversikan energi listrik menjadi energi mekanik adalah motor DC. Motor DC merupakan perangkat listrik yang sangat mudah untuk dikontrol. Apabila arus dialirkan melalui kumparan jangkar dari sebuah mesin DC dan kumparan medannya diberi penguatan maka akan timbul gaya *Lorentz* pada setiap sisi kumparan jangkar tersebut (Soemarwanto, 2010). Hal tersebut merupakan prinsip dasar kerja motor DC.

Besar gaya *Lorentz* yang ditimbulkan motor DC adalah:

$$F = B.I.l \text{ (dyne)} \quad (2.1)$$

Keterangan:

B = kerapatan medan magnetik (Gauss)

I = arus listrik (Ampere)

l = panjang konduktor (cm)

Akibat adanya gaya *Lorentz*, maka akan timbul kopel yang besarnya adalah:

$$\text{Kopel: } M = F.r \text{ (dyne-cm)} \quad (2.2)$$

Keterangan:

F = gaya *Lorentz* (dyne)

r = jari-jari jangkar (cm)

$$M = B.I.l.r \text{ (dyne-cm)} \quad (2.3)$$

Apabila jumlah: $M = z.B.I.l.r$ (dyne-cm), karena $B = \frac{\theta}{A}$.

Keterangan:

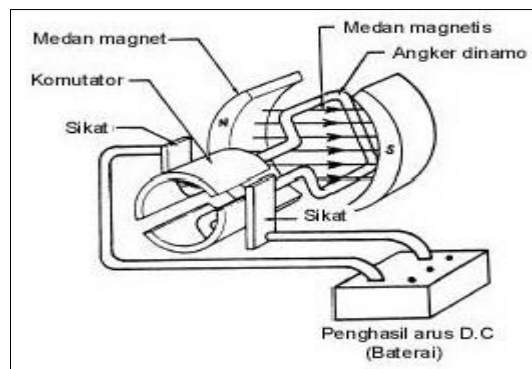
θ = fluksi tiap kutub (*Maxwell*)

A = luas penampang (cm²)

z = jumlah konduktor pada jangkar

maka: $M = z \cdot \frac{\theta}{A} \cdot I \cdot l \cdot r$ (dyne-cm) (2.4)

Motor DC ditunjukkan pada Gambar 2.3 membutuhkan catu daya tegangan yang searah pada kumparan medan. Catu tegangan tersebut akan menuju ke lilitan melalui sikat yang bersentuhan dengan komutator, dua segmen yang terhubung dengan dua ujung lilitan. Kumparan dalam satu lilitan yang berputar di antara medan magnet disebut dengan *Armature* (jangkar)



Gambar 2.16 Motor *Direct Current* (DC).

2.8 *Electric Fan*

Electric fan DC 12V adalah motor DC 12V yang telah dipasang baling-baling. Kuat hembusan angin yang digerakkan oleh motor DC bergantung pada jumlah baling-baling, desain dan RPM *electric fan* itu sendiri. Pada umumnya kipas ini telah dilengkapi dengan pelindung agar aman saat penggunaannya.



Gambar 2.17 *Electric Fan*

2.9 Pulse with Modulation (PWM)

Metode *Pulse Width Modulation* (PWM) adalah salah satu metode yang dapat digunakan untuk mengatur kecepatan dari motor DC. Kecepatan motor DC tergantung pada besarnya *duty cycle* yang diberikan pada motor DC tersebut. Pada sinyal PWM, frekuensi sinyal konstan sedangkan *duty cycle* bervariasi dari 0%-100%. Dengan mengatur *duty cycle*, akan diperoleh keluaran yang diinginkan (Alfarisi.R.,2011). *Duty cycle* adalah besarnya sinyal kontrol yang diberikan pada motor. Persamaan untuk perhitungan *duty cycle* ditunjukkan dalam persamaan 2.5 dengan T_{ON} adalah periode logika tinggi dan T adalah periode keseluruhan.

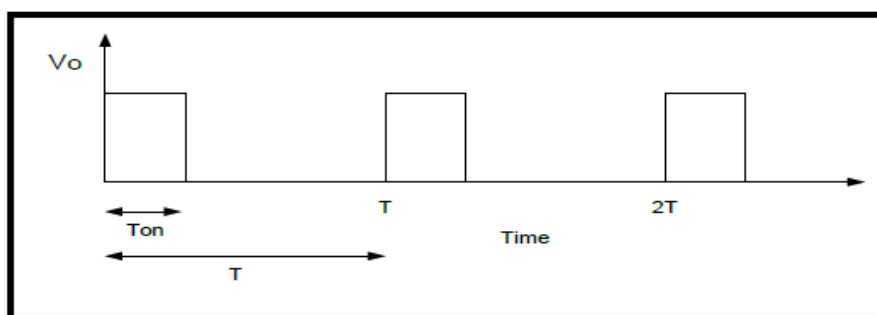
$$Duty\ cycle = \frac{T_{ON}}{T} \times 100\% \quad (2.5)$$

Keterangan:

T_{ON} = periode logika tinggi

T = periode keseluruhan

Sinyal PWM secara umum ditunjukkan oleh Gambar 2.6.



Gambar 2.18 Sinyal PWM.

Sumber: Ardyani, F. (2013)

2.10 High Current 30 A DC Motor Driver dengan Half-Bridge Infineon BTS 7960 chip

High Current 30 A DC Motor Driver (Gambar 2.7) adalah *driver* H-Bridge yang dirancang untuk dapat menghasilkan *drive* dengan 2 arah gerakan *clockwise* (cw) dan *counterclockwise* (ccw). Arus yang didapat juga kontinu hingga 30 A dan tegangan antara 5 hingga 27 Volt. Modul ini mampu untuk menggerakkan motor DC 12V dan dilengkapi

dengan rangkaian sensor arus beban. Rangkaian ini dihubungkan dengan mikrokontroler Arduino Mega 2560.



Gambar 2.19 *High Current 30 A Motor Driver.*

Fungsi masing-masing pin pada *Interface Header* dapat dilihat dalam Tabel 2.1.

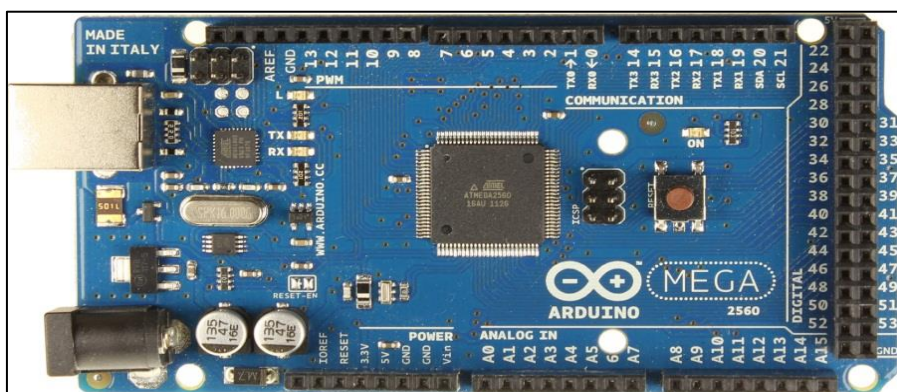
Tabel 2.1 Fungsi pin pada *Interface Header High Current 30 A DC Motor Driver (datasheet)*

No.	Pin	Fungsi
1.	RPWM	<i>Forward level atau input sinyal PWM, aktif HIGH</i>
2.	LPWM	<i>Reverse level atau input sinyal PWM, aktif LOW</i>
3.	R_EN	<i>Forward drive enable input, HIGH enable, LOW close</i>
4.	L_EN	<i>Reverse drive enable input, HIGH enable, LOW close</i>
5.	R_IS	<i>Forward drive-side current alarm output</i>
6.	L_IS	<i>Reverse drive-side current alarm output</i>
7.	VCC	<i>5V power input dari board arduino</i>
8.	GND	<i>Ground</i>

2.11 Mikrokontroler Arduino Mega 2560

Arduino Mega 2560 adalah *board* mikrokontroler berbasis ATmega 2560. Modul ini memiliki 54 digital *input/output* dimana 14 digunakan untuk PWM *output* dan 16 digunakan sebagai analog *input*, 4 untuk UART, 16 MHz osilator kristal, koneksi USB, *power jack*, ICSP *Header*, dan tombol *reset*. Modul ini memiliki segalanya yang dibutuhkan untuk merancang program mikrokontroler seperti kabel USB dan sumber daya melalui *adaptor* maupun *battery*.

Bahasa yang digunakan dalam modul arduino bukan bahasa *assembler* yang relatif sulit, tetapi bahasa C yang disederhanakan dengan bantuan pustaka-pustaka (*libraries*) arduino. Arduino juga menyederhanakan proses bekerja dengan mikrokontroler, dapat dilihat dengan Gambar 2.8.



Gambar 2.20 Arduino Mega 2560.

Sumber: *electroschematics.com*

Oleh karena itu, sehubungan dengan pembahasan untuk saat ini *software* arduino yang akan digunakan adalah *driver* dan IDE, walaupun masih ada beberapa *software* lain yang sangat canggih ditulis dengan menggunakan *Java*. IDE arduino terdiri dari :

- Editor program, sebuah *window* yang memungkinkan pengguna menulis dan mengedit program dalam bahasa *processing*.
- *Compiler*, sebuah modul yang mengubah kode program (bahasa *processing*) menjadi kode biner.
- *Uploader*, sebuah modul yang memuat kode biner dari komputer ke dalam memori di dalam papan arduino.

2.11.1 Daya

Arduino Mega 2560 dapat diaktifkan melalui koneksi USB atau dengan catu daya eksternal. Eksternal (non-USB) daya dapat berasal baik dari AC-ke adaptor-DC atau baterai. Adaptor ini dapat dihubungkan dengan menghubungkan *plug jack* pusat-positif ukuran 2.1mm konektor *power*. Ujung kepala dari baterai dapat dimasukkan ke dalam *Gnd* dan *V_{IN}* pin header dari konektor *power*. Kisaran kebutuhan daya yang disarankan untuk *board* Arduino Mega 2560 adalah 7 sampai dengan 12 Volt, jika diberi daya kurang dari 7 volt

kemungkinan pin 6V Arduino Mega 2560 dapat beroperasi tetapi tidak stabil. Kemudian jika diberi daya lebih dari 12V, regulator tegangan bisa panas dan dapat merusak *board*.

Pin listrik pada Arduino Mega 2560 adalah sebagai berikut:

1. V_{IN} adalah tegangan masukan kepada *board* arduino ketika itu menggunakan sumber daya eksternal (sebagai pengganti dari 5 volt koneksi USB atau sumber daya lainnya).
2. 5V adalah catu daya digunakan untuk daya mikrokontroler dan komponen lainnya.
3. 3v3 adalah sebuah pasokan 3,3 Volt dihasilkan oleh *regulator on-board*. Maksimum arus adalah 50 mA.
4. *GND* adalah sebuah *ground pin*.

2.11.2 Memori

Atmega2560 memiliki 256 KB (dengan 8 KB digunakan untuk bootloader), 8 KB dari SRAM dan 4 KB EEPROM.

2.11.3 Input & Output

Masing-masing dari 54 pin digital pada Arduino Mega 2560 dapat digunakan sebagai *input* ataupun *output*, dengan menggunakan fungsi *pinMode()*, *digitalWrite()*, dan *digitalRead()*, beroperasi dengan daya 5 Volt. Setiap pin dapat memberikan atau menerima maksimum 40 mA dan memiliki *internal pull-up* resistor (secara *default* terputus) dari 20-50 KOhms.

Selain itu, beberapa pin memiliki fungsi khusus:

1. *Serial*: 0(RX) dan 1(TX); *Serial 1*: 19(RX) dan 18(TX); *Serial 2*: 17(RX) dan 16(TX); *Serial 3*: 15(RX) dan 14(TX) digunakan untuk menerima (RX) dan mengirimkan (TX) TTL data *serial*. Pin ini dihubungkan ke pin yang berkaitan dengan *chip Serial ATmega8U2 USB-to-TTL*.
2. *Eksternal menyela*: 2(*interrupt 0*), 3(*interrupt 1*), 18(*interrupt 5*), 19(*interrupt 4*), 20(*interrupt 3*), 21(*interrupt 2*). Pin ini dapat dikonfigurasi untuk memicu *interrupt* pada nilai yang rendah, dengan batasan tepi naik atau turun, atau perubahan nilai.

3. *PWM*: 0 hingga 13. Menyediakan *output* PWM 8-bit dengan fungsi *analogWrite()*.
4. *SPI*: 53 (SS), 51 (MOSI), 50 (MISO), 52 (SCK). Pin ini mendukung komunikasi SPI menggunakan *SPI library*.
5. *LED*: 13. Ada *built-in* LED terhubung ke pin digital 13. Ketika pin bernilai nilai HIGH, LED *on*, ketika pin bernilai LOW, LED *off*.
6. *I²C*: 20(*SDA*) dan 21(*SCL*). Mendukung komunikasi *I²C* (*TWI*) menggunakan *Wire Library*.

Arduino Mega 2560 memiliki 16 masukan analog, berlabel A0 sampai dengan A15, yang masing-masing menyediakan 10 bit dengan resolusi (yaitu 1024 nilai yang berbeda). Selain itu, beberapa pin memiliki fungsi khusus :

1. *Aref*. Tegangan referensi (0 sampai 5V saja) untuk input analog. Digunakan dengan fungsi *analogReference()*.
2. *Reset*. Bawa baris ini LOW untuk me-*reset* mikrokontroler.

2.11.4 Komunikasi

Arduino memiliki sejumlah fasilitas untuk berkomunikasi dengan komputer, arduino lain, atau mikrokontroler lainnya. Atmega2560 menyediakan UART TTL (5V) untuk komunikasi serial, yang tersedia di pin digital 0 (RX) dan 1 (TX). Sebuah ATmega8U2 sebagai saluran komunikasi *serial* melalui USB dan sebagai *port virtual com* untuk perangkat lunak pada komputer. *Firmware* '8 U2 menggunakan *driver* USB standar COM, dan tidak ada *driver* eksternal yang diperlukan. Namun, pada *windows* diperlukan sebuah *file* inf. Perangkat lunak arduino terdapat monitor *serial* yang memungkinkan digunakan memonitor data tekstual sederhana yang akan dikirim ke atau dari board Arduino. LED RX dan TX di layar akan berkedip ketika data sedang dikirim melalui *chip* USB-to-serial dengan koneksi USB ke komputer (tetapi tidak untuk komunikasi serial pada pin 0 dan 1).

2.12 Kontroler

Salah satu komponen sistem pengaturan yang berfungsi mengolah sinyal umpan balik dan sinyal masukan acuan (*setpoint*) atau sinyal *error* mejadi sinyal kontrol. Sistem pengendalian dirancang untuk melakukan dan menyelesaikan tugas tertentu. Syarat utama sistem pengendalian adalah harus stabil. Di samping kestabilan mutlak, maka sistem harus memiliki kestabilan secara relatif, yakni tolak ukur kualitas kestabilan sistem dengan

menganalisis sampai sejauh mana batas-batas kestabilan sistem tersebut jika diberi gangguan (Ogata K.,1997). Selain itu analisis juga dilakukan untuk mengetahui bagaimana kecepatan sistem dalam merespon *input* dan bagaimana peredaman terhadap adanya lonjakan (*overshoot*).

Suatu sistem dikatakan stabil jika diberi gangguan maka sistem tersebut akan kembali ke keadaan *steady state* di mana *output* berada dalam keadaan tetap seperti tidak ada gangguan. Sistem dikatakan tidak stabil jika *output*nya berosilasi terus menerus ketika dikenai suatu gangguan. Karena suatu sistem pengendalian biasanya melibatkan penyimpanan energi, maka *output* sistem ketika diberi suatu *input*, tidak dapat mengikuti *input* secara serentak, tapi menunjukkan respon transien berupa suatu osilasi teredam sebelum mencapai *steady state*. Dalam sistem pengendalian terdapat 2 macam loop:

1. Pengendalian dengan Loop Terbuka

Sistem kontrol *loop* terbuka adalah sistem kontrol yang keluarannya tidak berpengaruh pada aksi pengontrolan. Jadi, pada sistem kontrol *loop* terbuka, keluaran tidak diukur atau diumpan balikan untuk dibandingkan dengan masukan.

2. Pengendalian dengan Loop Tertutup

Sistem kontrol *loop* tertutup adalah sistem kontrol yang keluarannya mempunyai pengaruh langsung pada aksi pengontrolan. Disebut juga sistem kontrol yang menggunakan umpan balik untuk memperkecil kesalahan sistem.

Untuk kontroler dengan aksi kontrol proporsional, hubungan antara keluaran kontroler $m(t)$ dan sinyal kesalahan penggerak $e(t)$ adalah:

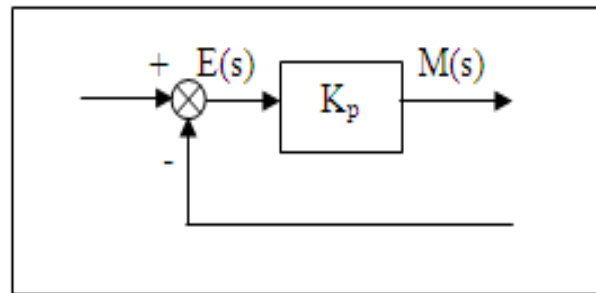
$$m(t) = K_p e(t)$$

atau, dalam besaran transformasi *Laplace*,

$$(M(s))/E(s) = K_p$$

dimana K_p adalah penguatan proporsional.

Apapun wujud mekanisme yang sebenarnya dan apapun bentuk daya penggeraknya, kontroler proporsional pada dasarnya merupakan penguat dengan penguatan yang dapat diatur (Ogata K.,1997). Diagram blok kontroler proporsional ditunjukkan dalam Gambar 2.21.



Gambar 2.21 Diagram Blok Kontroler Proporsional.

Sumber: Ogata K., (1997)

2.13 Jenis-jenis Kontroler

2.13.1 Kontroler Proporsional (P)

Kontroler proporsional adalah sebuah kontroler yang memiliki karakteristik mempercepat respon.

Hubungan antara keluaran kontroler $m(t)$ dan sinyal kesalahan penggerak $e(t)$ ditunjukkan pada persamaan 2.7:

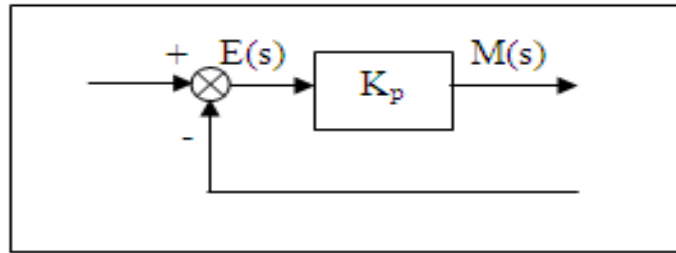
$$m(t) = K_p e(t) \quad (2.7)$$

atau, dalam besaran transformasi *Laplace* ditunjukkan pada persamaan 2.8:

$$\frac{M(s)}{E(s)} = K_p \quad (2.8)$$

dimana K_p adalah kepekaan proporsional atau penguatan.

Apapun wujud mekanisme yang sebenarnya dan apapun bentuk daya penggerakannya, kontroler proporsional pada dasarnya merupakan penguat dengan penguatan yang dapat diatur (Ogata K.,1997). Diagram blok kontroler proporsional ditunjukkan pada Gambar 2.22.



Gambar 2.22 Diagram Blok Kontroler Proporsional.

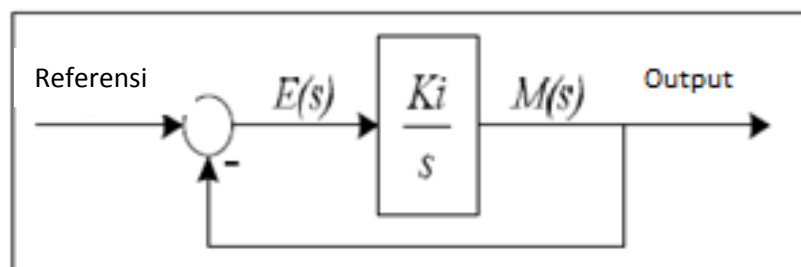
Sumber: Ogata K., (1997)

2.13.2 Kontroler Integral (I)

Kontroler integral memiliki kemampuan untuk mengurangi *offset* yang ditinggalkan oleh kontroler proporsional. Harga keluaran kontroler $m(t)$ diubah dengan laju yang sebanding dengan sinyal kesalahan penggerak $e(t)$. Persamaan kontroler Ki ditunjukkan dalam persamaan 2.9 (Ogata K.,1997).

$$\frac{dm(t)}{dt} = Kie(t) \quad (2.9)$$

Ki adalah penguatan integral. Gambar 2.23 menunjukkan diagram blok kontroler integral.



Gambar 2.23 Diagram Blok Kontroler Integral.

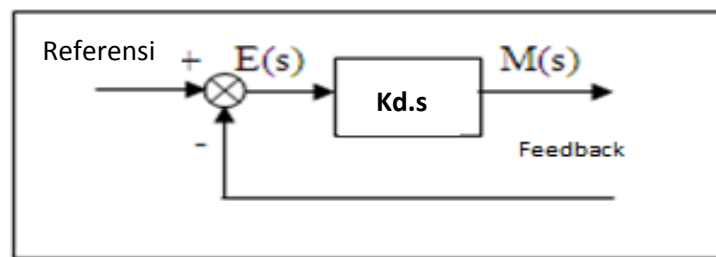
Sumber: Ogata K., (1997)

2.13.3 Kontroler Differensial (D)

Kontroler ini digunakan untuk memperbaiki atau mempercepat respon transien sebuah sistem kontrol dengan cara memperbesar *phase lead* terhadap penguatan kontrol dan mengurangi *phase lag* penguatan tersebut (Ogata K.,1997). Kontroler differensial tidak dapat mengeluarkan *output* bila tidak ada perubahan *input*, selain itu kontroler differensial tidak dapat digunakan untuk proses yang mengandung *noise*. Hubungan antara keluaran kontroler $m(t)$ dan sinyal kesalahan penggerak $e(t)$ dinyatakan pada persamaan 2.10.

$$\frac{M(s)}{E(s)} = Kd.s \quad (2.10)$$

Kd adalah penguatan differensial. Gambar 2.16 menunjukkan diagram blok kontroler differensial.



Gambar 2.24 Diagram Blok Kontroler Differensial.

Sumber: Ogata K., (1997)

2.13.4 Kontroler Proporsional Integral Differensial (PID)

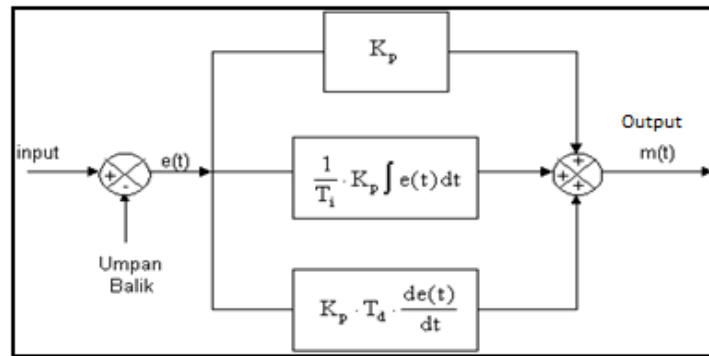
Gabungan aksi kontrol proporsional, integral, dan differensial mempunyai keunggulan dibandingkan dengan masing-masing dari tiga aksi kontrol tersebut. Masing – masing kontroler P, I, maupun D berfungsi untuk mempercepat reaksi sistem, menghilangkan *offset*, dan mendapatkan energi ekstra ketika terjadi perubahan *load*. Persamaan kontroler PID ini dapat dinyatakan dalam persamaan 2.11:

$$m(t) = Kp . e(t) + \frac{Kp}{Ti} . e(t)dt + Kp.Td \frac{de(t)}{dt} \quad (2.11)$$

Dalam transformasi *Laplace* dinyatakan dalam persamaan 2.12:

$$\frac{M(s)}{E(s)} = Kp \left(1 + \frac{1}{Ti.s} + Td.s \right) \quad (2.12)$$

Ti adalah waktu integral dan Td adalah waktu derivatif. Gambar 2.25 menunjukkan diagram blok kontroler PID.



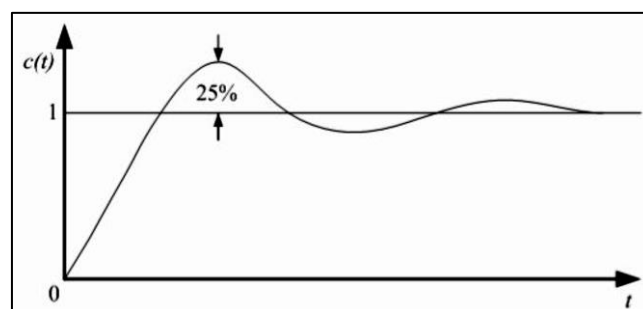
Gambar 2.25 Diagram Blok Kontroler PID.

Sumber: Ogata K., (1997)

2.14 Ziegler Nichols

Ziegler dan Nichols mengemukakan aturan-aturan untuk menentukan nilai dari *gain* proporsional K_p , waktu integral T_i , dan waktu derivatif T_d berdasarkan karakteristik respon transien dari *plant* yang diberikan. Penentuan parameter kontroler PID atau penalaan kontroler PID tersebut dapat dilakukan dengan bereksperimen dengan *plant* (Ogata, K., 1997).

Terdapat dua metode yang disebut dengan aturan penalaan *Ziegler-Nichols*, pada kedua metode tersebut memiliki tujuan yang sama yaitu untuk mencapai 25% *maximum overshoot* pada respon *unit step*, seperti ditunjukkan dalam Gambar 2.26.

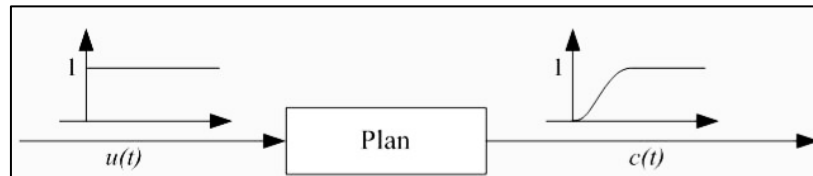


Gambar 2.26 Kurva Respon *Unit Step* yang Menunjukkan 25% *Maximum Overshoot*.

Sumber: Ogata K., (1997)

a. Metode Pertama

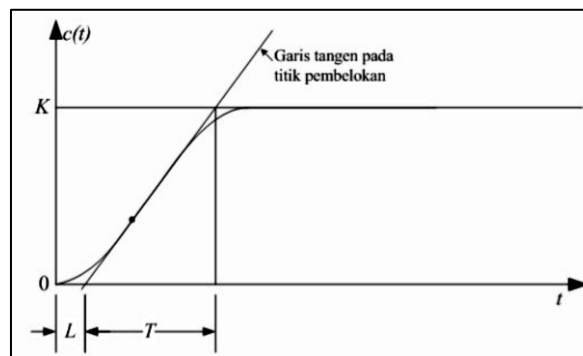
Metode pertama atau sering disebut metode kurva reaksi, respon dari *plant* dapat dapat diperoleh secara eksperimental dengan masukan berupa *unit step*, seperti yang ditunjukkan dalam Gambar 2.27.



Gambar 2.27 Respon *Plant* Terhadap Masukan Berupa *Unit Step*.

Sumber: Ogata K., (1997)

Jika dalam *plant* tersebut minimal tidak mengandung unsur integrator atau *dominan complex-conjugate poles*, maka kurva respon *unit step* berbentuk seperti huruf S, seperti dalam Gambar 2.20 jika respon tidak memberikan bentuk kurva S, maka metode ini tidak berlaku (Ogata, K., 1997).



Gambar 2.28 Kurva Respon yang Berbentuk S.

Sumber: Ogata, K., 1997

Kurva berbentuk S tersebut dapat dikarakteristikan menjadi dua konstanta yaitu waktu mati L dan waktu tunda T . Waktu mati dan waktu tunda ditentukan dengan menggambar sebuah garis tangent pada titik pembelokan dari kurva S, dan menentukan perpotongan antara garis tangent dengan sumbu waktu t dan sumbu $c(t) = K$, seperti yang telah ditunjukkan dalam Gambar 2.28 Fungsi alih $C(s)/U(s)$ dapat dilakukan pendekatan dengan sistem orde satu dengan persamaan sebagai berikut:

$$\frac{C(s)}{U(s)} = \frac{Ke^{-Ls}}{Ts+1} \quad (2.13)$$

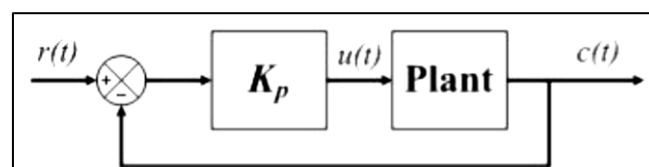
Ziegler dan *Nichols* menyarankan untuk menentukan nilai-nilai dari K_p , T_i dan T_d berdasarkan pada formula yang ditunjukkan dalam Tabel 2.2.

Tabel 2.2 Aturan Penalaan *Ziegler-Nichols* Berdasarkan Respon *Unit Step* dari *Plant*
(Sumber: Ogata, K., 1997)

Typo Kontroler	K_p	T_i	T_d
P	$\frac{T}{L}$	∞	0
PI	$0,9 \frac{T}{L}$	$\frac{L}{0,3}$	0
PID	$1,2 \frac{T}{L}$	$2L$	$0,5 L$

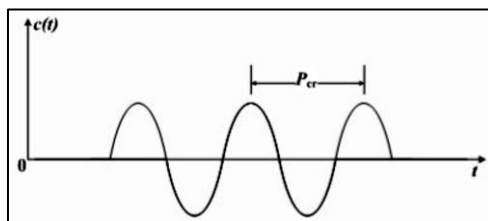
b. Metode Kedua

Dalam metode kedua *ziegler-nichols*, mula-mula yang dilakukan adalah membuat $T_i = \infty$ dan $T_d = 0$. Kemudian hanya dengan menggunakan tindakan kontrol proporsional, harga ditingkatkan dari nol ke suatu nilai kritis K_{cr} , disini mula-mula keluaran memiliki osilasi yang berkesinambungan (Jika keluaran tidak memiliki osilasi berkesinambungan untuk nilai K_p manapun yang telah diambil, maka metode ini tidak berlaku). Dari keluaran yang berosilasi secara berkesinambungan, penguatan kritis K_{cr} dan periode P_{cr} dapat ditentukan. Diagram blok sistem *loop* tertutup dengan kontroler proporsional dapat dilihat dalam Gambar 2.21 dan untuk osilasi berkesinambungan dengan periode P_{cr} dapat dilihat dalam Gambar 2.22 *Ziegler* dan *Nichols* menyarankan penyetelan nilai parameter K_p , T_i , T_d dan berdasarkan rumus yang diperlihatkan dalam Tabel 2.3.



Gambar 2.29 Sistem *Loop* Tertutup dengan Kontroler Proporsional.

Sumber: Ogata K., (1997)



Gambar 2.30 Osilasi Berkesinambungan dengan Periode P_{cr} .

Sumber: Ogata K., (1997)

Tabel 2.3 Aturan Dasar Ziegler-Nichols Berdasarkan *Critical Gain K_{cr}* dan *Critical Period P_{cr}* (Sumber: Ogata, K. 1997)

Tipe Kontroler	K_p	T_i	T_d
P	$0.5 K_{cr}$	∞	0
PI	$0.45 K_{cr}$	$\frac{1}{1.2} P_{cr}$	0
PID	$0.60 K_{cr}$	$0.5 P_{cr}$	$0.125r$

2.15 Transformasi Z

Dalam suatu sistem yang berulang, transformasi *Laplace* dapat digunakan dalam sebuah analisis maupun desain. Sedangkan, transformasi Z memiliki fungsi untuk dapat dimanfaatkan dalam menganalisa sistem yang termodelkan dalam waktu diskrit dari persamaan differensial. Sebuah transformasi ditetapkan sebagai deret angka yang berurutan. Fungsi $E(z)$ ditetapkan sebagai sebuah *power series* dalam z^{-k} dengan koefisien yang sama untuk nilai dari deret angka $\{e(k)\}$. Transformasi ini disebut sebagai transformasi Z, yang kemudian diungkapkan dalam bentuk persamaan 2.13.

$$E(z) = Z[\{E(k)\}] = e(0) + e(1)z^{-1} + e(2)z^{-2} + \dots \quad (2.13)$$

$$\text{Dengan } e(k) = z^{-1} \{E(z)\} = \frac{1}{2\pi f} \oint_{\Gamma} E(z) z^{k-1} dz, \quad f = \sqrt{-1}$$

Di mana $z(\dots)$ mengindikasikan operasi transformasi Z dan $Z^{-1}(\dots)$ mengindikasikan *invers* dari transformasi z. $E(z)$ dalam Persamaan 2.13 dapat dituliskan dalam notasi yang lebih sederhana seperti pada persamaan berikut:

$$E(z) = Z[\{e(k)\}] = \sum_{k=0}^{\infty} e(k)z^{-k} \quad (2.14)$$

Transformasi Z terdefiniskan untuk beberapa deret angka $\{e(k)\}$, dan mungkin digunakan dalam menganalisis dari beberapa jenis sistem yang diuraikan oleh persamaan differensial *time-invariant* linier.

2.16 Diskritisasi

Ada beberapa cara dalam menentukan proses diskritisasi diantaranya adalah *backward difference*, *forward difference*, dan *bilinear transform*. Semua metode adalah metode pendekatan (*approximations*), sehingga hasil yang diperoleh tidak akan sama dengan bentuk analognya. Hal ini disebabkan karena bentuk diskrit akan ada bagian informasi yang hilang, karena formula matematika yang digunakan dalam proses diskritisasi juga diturunkan berdasarkan pendekatan yang lebih mudah digunakan.

Menurut teori *sampling Nyquist* dan *Shannon*, jika fungsi waktu $e(t)$ tidak terdapat unsur frekuensi yang lebih tinggi 1 Hz, dalam hal ini nilai dari nilai $e(t)$ dapat ditentukan dengan memberikan titik *sampling* berjarak $1/2f_0$ detik terpisah.

Berikut tahapan diskritisasi yang dapat digunakan:

1. Tulis algoritma analog bentuk transformasi *Laplace*.
2. Lakukan diskritisasi menjadi bentuk transformasi Z dengan mengganti operator s dengan menggunakan salah satu dari titik metode diskritisasi yaitu:

- *Back difference*:

$$s = \frac{1-z^{-1}}{T_s} \quad (2.15)$$

- *Forward difference*:

$$s = \frac{1-z^{-1}}{T_s z^{-1}} \quad (2.16)$$

- *Bilinear transform*:

$$s = \frac{2(1-z^{-1})}{T_s(1+z^{-1})} \quad (2.17)$$

Dimana T_s merupakan waktu *sampling*.

Hingga tahap 2, algoritma sudah didapat dalam bentuk diskrit yang dinyatakan dalam bentuk transformasi Z. Namun, pada implementasinya bentuk transformasi Z tersebut perlu diubah menjadi bentuk *time domain*, yaitu dengan mengubah operator z^n menjadi n kali waktu *delay* (z^{-1} berarti 1 kali waktu *delay*, z^{-2} berarti 2 kali waktu *delay* dan seterusnya).

2.17 Parameter Performansi Sistem

Parameter-parameter dibagi menjadi lima (performansi sistem) antara lain:

- a. Waktu penetapan (*Settling time*), T_s : waktu yang diperlukan kurva respon untuk mencapai dan menetap dalam daerah di sekitar nilai akhir yang ukurannya ditentukan dengan presentase mutlak dari nilai akhir (biasanya 5% atau 2%). Waktu penetapan ini dihubungkan dengan konstanta waktu terbesar dari sistem pengontrolan. Kriteria presentase kesalahan yang akan digunakan ditentukan dari sasaran desain.
- b. Waktu tunda (*Delay time*), T_d : waktu yang diperlukan respon untuk mencapai setengah nilai akhir pada saat lonjakan yang pertama kali.
- c. Waktu naik (*Rise time*), T_r : waktu yang diperlukan respon untuk naik dari 10 sampai 90%, 5 sampai 95%, atau 0 sampai 100% dari nilai akhir. Untuk sistem orde dua redaman kurang (*Underdamped*) digunakan waktu naik 0 hingga 100% dan untuk sistem redaman lebih (*Overdamped*) digunakan waktu naik 10 hingga 90%.
- d. Waktu puncak (*Peak time*), T_p : waktu yang diperlukan respon untuk mencapai puncak lewatan (lonjakan maksimal) yang pertama kali.
- e. Lewatan maksimum (*Maximum overshoot*), M_p : nilai puncak maksimum kurva respon yang diukur dari satu. Jika nilai keadaan mantap respon tidak sama dengan satu, maka dapat digunakan persen lewatan maksimum.

BAB III

METODE PENELITIAN

Metode penelitian pada dasarnya merupakan cara ilmiah untuk mendapatkan informasi dengan tujuan dan manfaat tertentu. Dalam menyelesaikan masalah dan merealisasikan tujuan penelitian yang terdapat di bab pendahuluan maka diperlukan langkah-langkah untuk menyelesaikan masalah tersebut. Metode penelitian pada skripsi ini meliputi:

3.1 Perancangan blok diagram sistem

3.2 Spesifikasi desain

3.3 Karakterisasi setiap blok

Karakterisasi setiap blok dilakukan untuk mempermudah analisis sistem. Karakterisasi dibagi menjadi beberapa bagian, yaitu:

3.3.1 Karakterisasi Termokopel Tipe-K

3.3.2 Karakterisasi *Driver Motor (High Current 30 A DC Motor Driver)*

3.3.3 Karakterisasi *Electric Fan*

3.4 Pembuatan perangkat keras

3.4.1 Desain media pendingin mesin

3.4.2 Prinsip kerja alat

3.4.3 Rangkaian catu daya

3.4.4 Konfigurasi pin Arduino Mega 2560

3.4.5 Konfigurasi pin *driver motor*

3.5 Perancangan algoritma

3.5.1 Penentuan respon karakteristik pengontrolan suhu

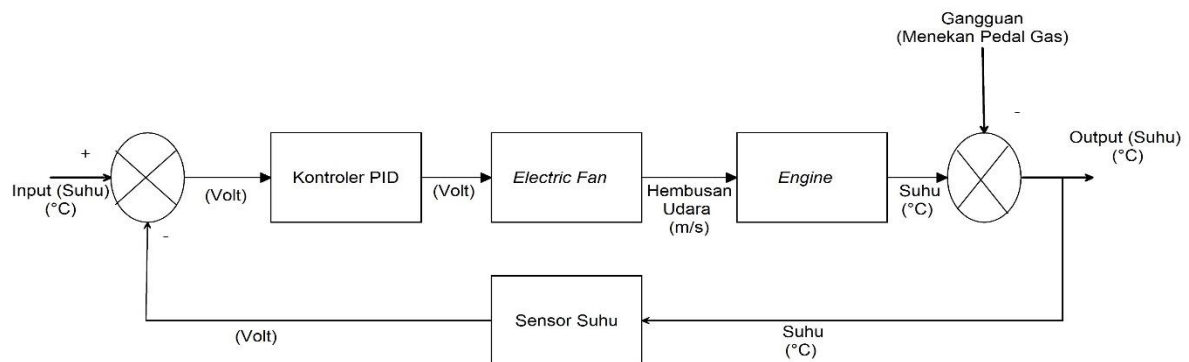
3.5.2 Parameter kontroler PID dengan metode *Ziegler-nichols*

3.5.3 Diskritisasi

3.5.4 *Flowchart* Program

3.1 Perancangan Blok Diagram Sistem

Pada perancangan alat diperlukan perancangan diagram blok sistem yang dapat menjelaskan sistem secara garis besar dan diharapkan alat dapat bekerja sesuai dengan yang diinginkan. Blok diagram sistem dapat dilihat pada gambar 3.1. pada gambar 3.1 merupakan diagram blok pengontrolan suhu pada radiator mesin mobil. Input yang digunakan pada sistem ini adalah suhu, dengan menggunakan kontroler PID, actuator yang digunakan adalah *electric fan* DC 12V, plant dari sistem ini adalah suhu mesin mobil, gangguan pada sistem ini adalah naiknya temperatur mesin dikarenakan menekan pedal gas pada rpm tertentu selama waktu yang ditentukan, sensor yang digunakan pada sistem ini adalah Termokopel Tipe-K dan output dari sistem ini juga berupa suhu.



Gambar 3.1 Blok Diagram Sistem Pendingin Mesin Mobil
Sumber: Perancangan

Keterangan :

- Input dari sistem ini yaitu suhu dengan nilai 86°C yang didapat dari suhu optimal mesin mobil.
- Kontroler berfungsi sebagai pembanding nilai masukan dan keluaran sistem dan menghasilkan sinyal kontrol untuk aktuator. Kontroler yang digunakan adalah kontroler PID dengan menggunakan metode Ziegler Nichols I
- Aktuator pada sistem ini berupa *Electric Fan*. Aktuator berfungsi untuk menggerakkan atau mengontrol sebuah mekanisme atau sistem.
- Plant dari sistem ini adalah radiator mesin mobil. Plant sendiri adalah objek fisik yang dikontrol. Sistem ini mengontrol suhu yang terdapat pada radiator mesin mobil pada suhu 86°C .

- Gangguan pada sistem ini berupa menekan pedal gas pada rpm dan kurun waktu tertentu.
- Output dari sistem ini berupa suhu mesin mobil dalam satuan derajat Celcius ($^{\circ}\text{C}$)
- Sensor yang digunakan dalam sistem ini adalah sensor termokopel tipe-K yang diletakkan pada selang radiator bagian atas. Sensor ini berfungsi sebagai *feedback* pada sistem.

3.2 Spesifikasi Desain

Desain yang diinginkan pada perancangan pengontrolan suhu mesin mobil yaitu:

1. Kendaraan yang digunakan adalah Suzuki Katana. Kendaraan ini digunakan dalam sistem ini karena kerap kali mengalami kenaikan suhu yang cukup signifikan dan juga mobil ini beredar cukup banyak di Indonesia
2. *Electric fan* yang digunakan mempunyai spesifikasi 12v – 80 W – 6A, diameter 27cm ketebalan 7cm, 2000rpm
3. Pemasangan *electric fan* ini diletakkan pada bagian depan dari radiator mobil.
4. Sensor suhu yang digunakan adalah Termokopel Tipe-K. sensor suhu ini digunakan karena *waterproof*, tersedia dengan berbagai macam probe dan mampu menerima suhu yang tinggi yaitu sampai dengan 400°C
5. Termokopel tipe-K membutuhkan penguat sinyal agar suhu dapat dibaca oleh manusia maupun mesin. Alat penguat suhu ini adalah Max 6675. Max 6675 telah didesain agar dapat menguatkan sinyal dari Termokopel tipe-K
6. Pemasangan sensor suhu Termokopel tipe-K ini diletakkan pada selang radiator mobil bagian atas.

3.3 Karakterisasi Setiap Blok

3.3.1 Karakterisasi Sensor Termokopel Tipe-K

Karakterisasi Termokopel tipe-K dilakukan untuk mengetahui karakter atau *gain* sensor tersebut. Hal tersebut diperoleh dengan mengamati keluaran pembacaan sensor suhu tersebut kemudian dibandingkan dengan Termometer Laboratorium.

Peralatan yang digunakan terdiri atas:

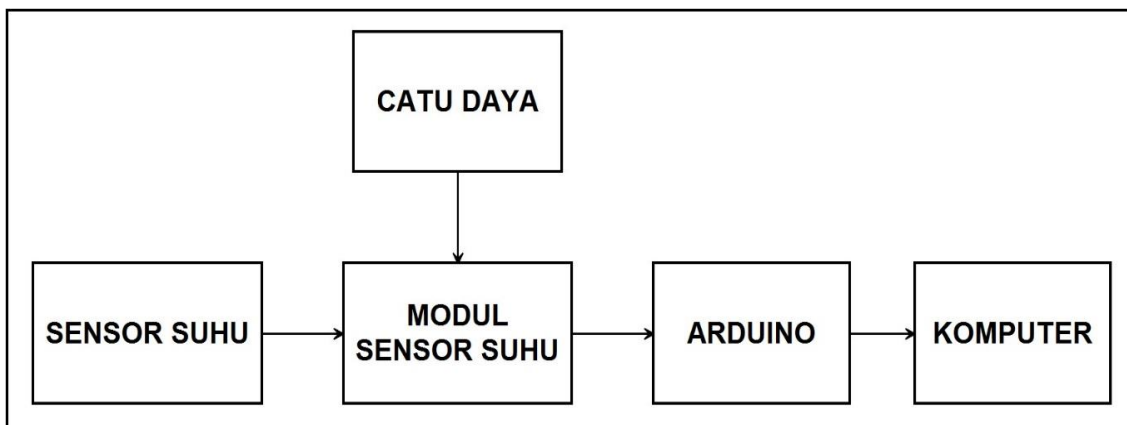
1. *Termokopel Tipe-K*
2. *Max6675*

3. Termometer Laboratorium
4. Komputer/Laptop
5. Arduino Mega 2560 (Mikrokontroler)
6. *Software* Arduino 1.6.7.
7. Beberapa gelas air dengan suhu yang berbeda
8. *Power supply* DC
9. Kabel penghubung

Langkah pengujian meliputi:

1. Merangkai *driver* Termokopel tipe-K dengan *Max6675* dengan cara menghubungkan kutub positif sensor ke kutub positif *Max6675* dan kutub negatif ke kutub negatif. Lalu menghubungkan *V+* ke pin 5V pada arduino, *GND* ke pin *GND* pada arduino, *thermoDo* ke pin digital 8, *thermoCs* ke pin digital 7, *thermoCLK* ke pin digital 6
2. Menghubungkan rangkaian *driver* Termokopel tipe-K dan arduino ke komputer/laptop.
3. Membuat program arduino yang sesuai dengan pengujian Termokopel tipe-K
4. Memasukkan sensor suhu Termokopel tipe-K dengan Termometer Laboratorium ke dalam beberapa gelas yang diisi air dengan suhu yang berbeda
5. Membandingkan data antara sensor Termokopel tipe-K dengan Termometer Laboratorium

Blok prosedur pengujian karakterisasi *Termokopel Tipe-K* dapat dilihat pada gambar 3.2.



Gambar 3.2 Blok Pengujian Termokopel tipe-K.
Sumber: Perancangan

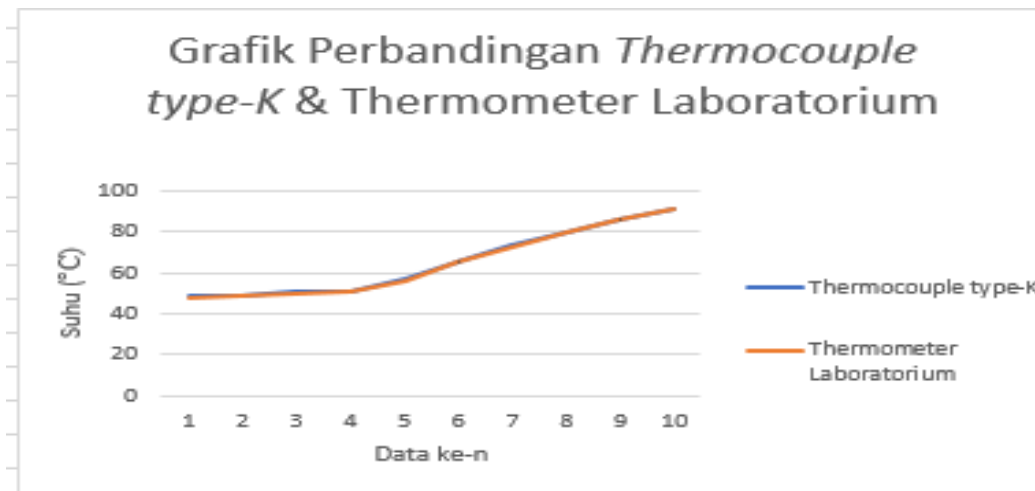
Data hasil pengujian pembacaan sensor suhu *Termokopel Tipe-K* terhadap pembacaan sensor suhu Termometer Laboratorium ditunjukkan dalam tabel 3.1. pada table tersebut terdapat pembacaan dari sensor suhu Thermometer Laboratorium sebagai acuan. Sedangkan terlihat pula pembacaan dari sensor Termokopel Tipe-K. kemudian *error* yang dihasilkan merupakan perbandingan dari pembacaan tersebut.

Tabel 3.1 Hasil Pengujian Termokopel Tipe-K

No.	<i>Termokopel</i> Tipe-K	Thermometer Laboratorium	Error
1	48.5	48	1.04
2	48.75	49	0.51
3	50.5	50	1.04
4	51.25	51	0.48
5	56.5	56	1.04
6	65.5	65	0.48
7	73.5	73	0.48
8	80	80	0
9	85.75	86	0.48
10	91	91	0
Rata-rata			0.555

Berdasarkan hasil pengujian dalam Tabel 3.1, rata-rata *error* antara suhu pembacaan dari Termokopel Tipe-K dengan pembacaan oleh Termometer Laboratorium setelah sebesar 0.555%. Grafik perbandingan suhu oleh Termometer Laboratorium dengan pembacaan Termokopel Tipe-K dapat dilihat dalam Gambar 3.2.

Error yang didapatkan haruslah sangat kecil, dikarenakan hasil pembacaan antara Thermometer Laboratorium dan Termokopel tipe-K tidak terpaut jauh. Dalam sistem ini membutuhkan umpan balik yang akurat dengan kepekaan yang tidak relatif tinggi, karena sistem akan dapat lebih stabil.



Gambar 3.3 Grafik Hasil Pengujian Sensor Termokopel Tipe-K.

Sumber: Perancangan

Dari grafik tersebut didapatkan *gain* sensor Termokopel tipe-K adalah:

$$m = \frac{y_{0.5} - y_{0.4}}{x_{0.5} - x_{0.4}} = \frac{50.5 - 48.75}{3 - 2} = 1.75$$

3.3.2 Karakterisasi *Driver Motor (High Current 30 A DC Motor Driver)*

Karakterisasi *driver motor (High Current 30 A DC Motor Driver)* dilakukan untuk mengetahui karakter atau *gain* dari *driver motor* tersebut. Hal tersebut diperoleh dengan mengamati tegangan keluaran *driver motor* terhadap perubahan *duty cycle input driver motor*.

Peralatan yang digunakan terdiri atas:

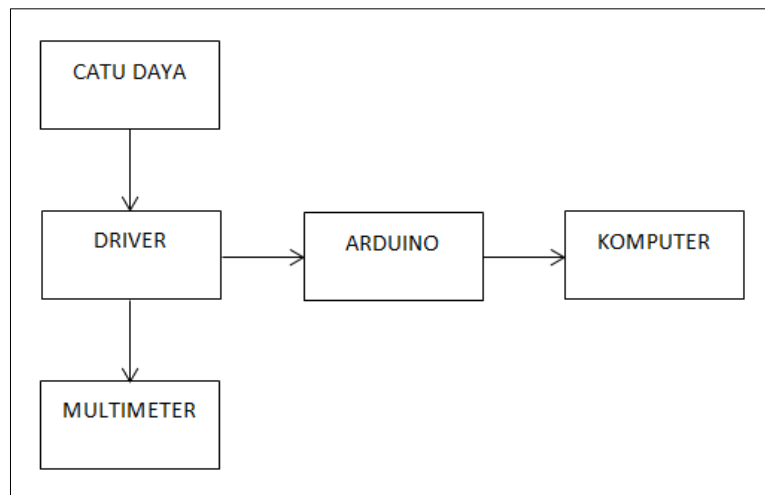
1. *Driver motor (High Current 30 A DC Motor Driver)*
2. Komputer/laptop
3. Arduino Mega 2560 (Mikrokontroler)
4. *Software* Arduino 1.6.7.
5. Multimeter digital CD800a
6. *Power supply* DC
7. Kabel penghubung

Langkah pengujian meliputi:

1. Merangkai *driver motor* ke Arduino Mega 2560 dengan cara menghubungkan PWM ke pin PWM arduino, *enable driver motor* ke pin digital arduino, *Vcc* ke pin 5V arduino, dan *GND* ke pin *GND* arduino.

2. Menghubungkan masukan sumber dari *power supply* DC ke masukan *driver* dan keluaran *driver* M+ ke (+) multimeter CD800a dan *driver* M- ke (-) multimeter CD800a.
3. Menghubungkan rangkaian *driver motor* dan arduino ke komputer/laptop.
4. Membuat program arduino yang sesuai dengan pengujian *driver motor* pada *software* Arduino 1.6.7.
5. Masukkan *duty cycle* secara berurutan mulai dari 0% hingga 100% dan catat data tegangan yang keluar dari multimeter CD800a.

Blok prosedur pengujian karakterisasi *driver motor* (*High Current 30 A DC Driver Motor*) dapat dilihat pada Gambar 3.4.



Gambar 3.4 Blok Pengujian *Driver Motor* (*High Current 30 A DC Motor Driver*).

Sumber: Perancangan

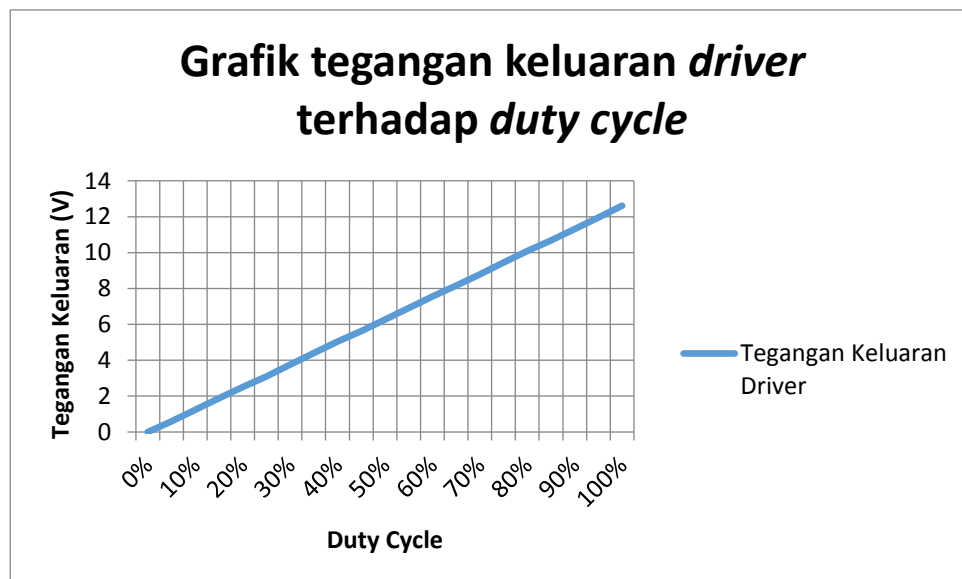
Data hasil pengujian tegangan keluaran *driver motor* (Volt) terhadap *duty cycle input driver motor* ditunjukkan dalam Tabel 3.2. Pada tabel tersebut terdapat *duty cycle* sebagai *input driver*. Sedangkan, *output driver* berupa tegangan yang diukur dengan menggunakan multimeter digital.

Tabel 3.2 Hasil Pengujian *Driver Motor (High Current 30 A DC Motor Driver)*.

No.	Duty Cycle (%)	Tegangan Keluaran Driver (V)	No.	Duty Cycle (%)	Tegangan Keluaran Driver (V)
1.	0	0	12.	55	6.91
2.	5	0.59	13.	60	7.55
3.	10	1.24	14.	65	8.15
4.	15	1.88	15.	70	8.79
5.	20	2.51	16.	75	9.44
6.	25	3.1	17.	80	10.08
7.	30	3.75	18.	85	10.68
8.	35	4.39	19.	90	11.32
9.	40	5.03	20.	95	11.96
10.	45	5.63	21.	100	12.61
11.	50	5.27	22.	-	-

Berdasarkan Tabel 3.2 dapat disimpulkan bahwa semakin besar nilai *duty cycle driver motor (High Current 30 A DC Motor Driver)* maka akan semakin besar pula nilai tegangan keluaran *driver motor (High Current 30 A DC Motor Driver)*.

Grafik hasil pengujian tegangan keluaran *driver motor* (volt) terhadap *duty cycle input driver motor* ditunjukkan dalam Gambar 3.5.

**Gambar 3.5** Grafik hasil pengujian *Driver Motor (High Current 30 A DC Motor Driver)*.

Sumber: Perancangan

Dari grafik tersebut didapatkan *gain driver motor* adalah:

$$m = \frac{y_{0.5} - y_{0.4}}{x_{0.5} - x_{0.4}} = \frac{5.27 - 5.03}{50 - 40} = 0.024$$

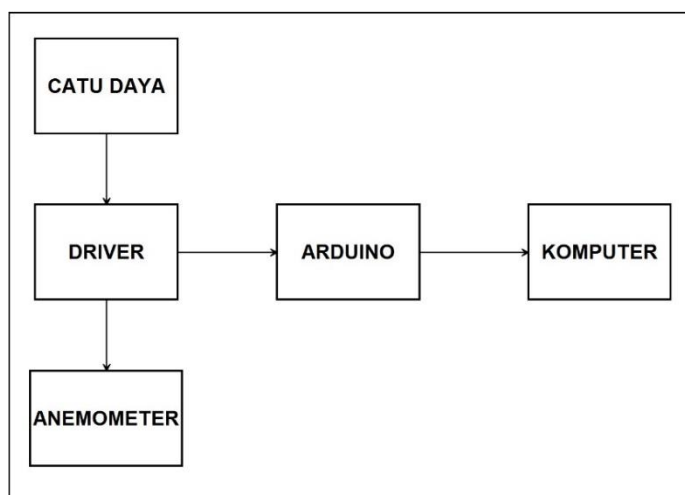
3.3.3 Karakterisasi *Electric Fan*

Karakterisasi sensor *Electric Fan* dilakukan untuk mengetahui karakter sensor tersebut. Hal tersebut diperoleh dengan mengamati *air flow* yang dihembuskan oleh *Electric Fan* terhadap perubahan *duty cycle*. Peralatan yang digunakan terdiri atas:

1. *Electric Fan*
2. *Driver motor (High Current 30 A DC Motor Driver)*
3. Komputer/laptop
4. Arduino Mega 2560 (Mikrokontroler)
5. *Software Arduino 1.6.7.*
6. Anemometer
7. Kabel penghubung

Langkah pengujian meliputi:

1. Merangkai *driver motor* ke Arduino Mega 2560 dengan cara menghubungkan PWM ke pin PWM arduino, *enable driver motor* ke pin digital arduino, *Vcc* ke pin 5V arduino, dan *GND* ke pin *GND* arduino.
2. Menghubungkan masukan sumber dari *power supply* DC ke masukan *driver* dan keluaran *driver* M+ ke (+) motor pompa mini DC 12V dan *driver* M- ke (-) motor pompa mini DC 12V.
3. Menghubungkan rangkaian motor pompa dan arduino ke komputer/laptop
4. Membuat program yang sesuai dengan pengujian motor DC 12V pada *software Arduino 1.6.7.*
5. Masukkan *duty cycle* secara berurutan mulai dari 0% hingga 100% dan catat data *air flow* yang keluar dari *electric fan*.



Gambar 3.6 Blok Pengujian *Electric fan* DC 12V
Sumber: Perancangan

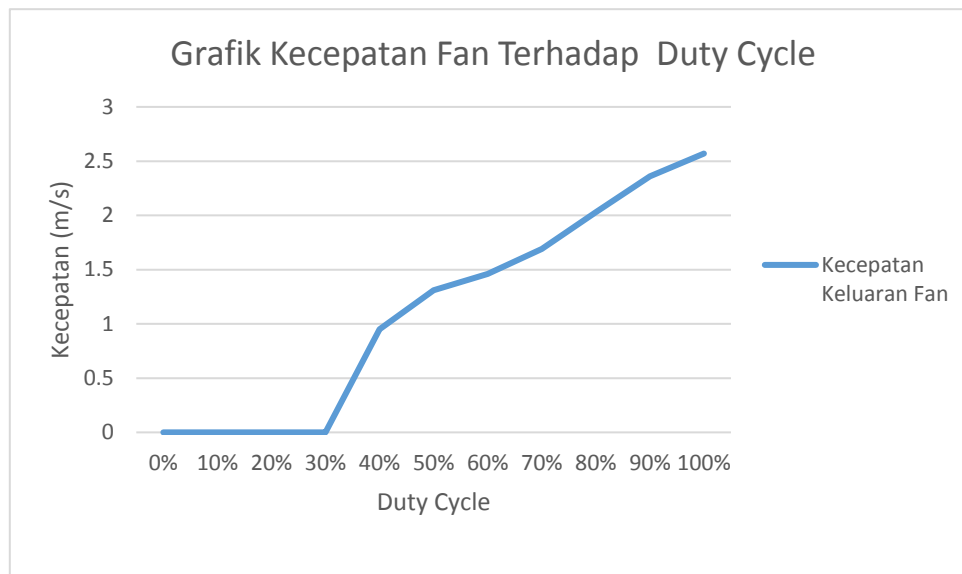
Data hasil pengujian tegangan keluaran *driver motor* (Volt) terhadap *duty cycle input driver motor* ditunjukkan dalam tabel 3.3. Pada tabel tersebut terdapat *duty cycle* sebagai *input driver*. Sedangkan, *output driver* berupa tegangan yang diukur dengan menggunakan anemometer digital

Tabel 3.3 Hasil Pengujian *Electric fan* DC 12V

No.	Duty Cycle (%)	m/s	No.	Duty Cycle (%)	m/s
1.	0	0	12.	55	1.46
2.	5	0	13.	60	1.55
3.	10	0	14.	65	1.69
4.	15	0	15.	70	1.80
5.	20	0	16.	75	1.96
6.	25	0	17.	80	2.03
7.	30	0	18.	85	2.16
8.	35	0.95	19.	90	2.36
9.	40	1.23	20.	95	2.44
10.	45	1.31	21.	100	2.57
11.	50	1.39	22.	-	-

Berdasarkan tabel 3.2 dapat disimpulkan bahwa semakin besar nilai *duty cycle driver motor* (*High Current 30 A DC Motor Driver*) maka akan semakin besar pula nilai kecepatan putaran *Electric Fan*.

Grafik hasil pengujian tegangan keluaran *driver motor* (volt) terhadap *duty cycle input driver motor* ditunjukkan dalam Gambar 3.7



Gambar 3.7 Grafik Hasil Kecepatan Fan Terhadap *Duty Cycle*
Sumber: Perancangan

Dari grafik tersebut didapatkan *gain Electric Fan* adalah:

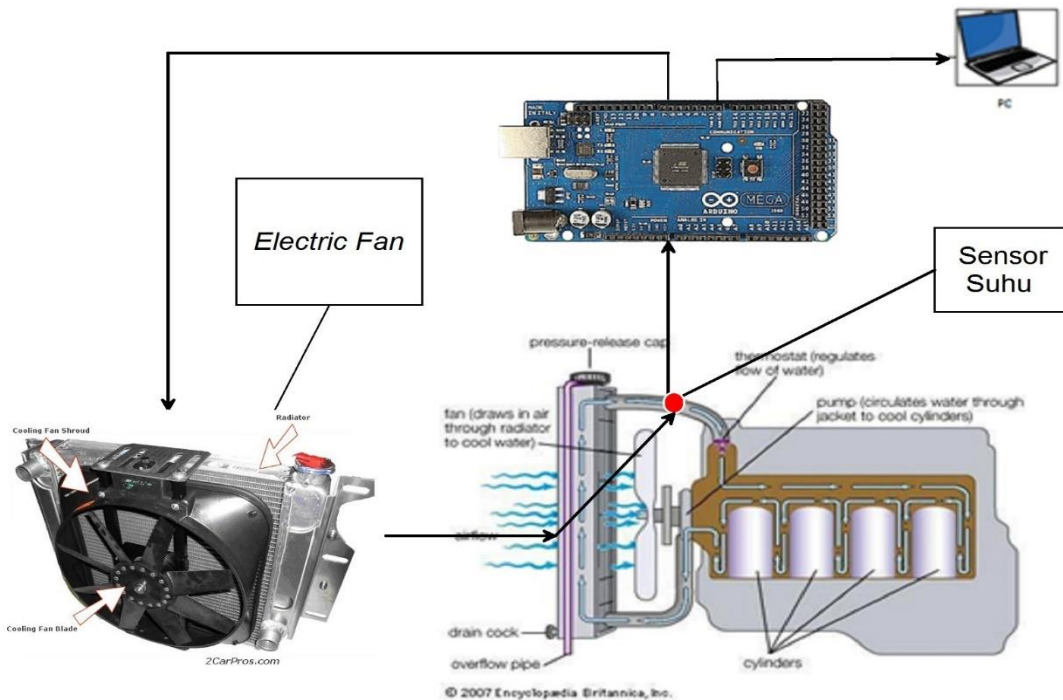
$$m = \frac{y_{0.8} - y_{0.7}}{x_{0.8} - x_{0.7}} = \frac{1.96 - 1.80}{80 - 70} = 0.016$$

3.4 Pembuatan Perangkat Keras

3.4.1 Desain Media

Sistem pendingin udara ini menggunakan *electric fan* yang dikontrol oleh Arduino dengan metode *Ziegler Nichols 1*. Dalam selang radiator bagian atas terdapat sensor suhu Termokopel Tipe-K yang akan terhubung ke Arduino. Besarnya suhu yang dibaca oleh sensor suhu akan dibaca oleh Arduino, apabila suhu mesin mencapai suhu 90°C maka *electric fan* akan aktif. Kipas akan padam apabila suhu mesin dibawah 90°C. Suhu optimal mesin adalah 86°C, dalam penggunaannya suhu mesin dapat mengalami kenaikan suhu akan tetapi masih belu mencapai

ambang batas *Overheat*, maka dari itu *Electric Fan* menyala pada 90°C . Ambang batas overheat adalah 100°C . Pembacaan suhu akan dilakukan terus menerus selama mesin mobil dalam keadaan menyala. Seluruh sistem ini dapat dipantau oleh PC.



Gambar 3.8 Desain Alat
Sumber: Perancangan

3.4.2 Prinsip Kerja Alat

1. Sistem diberi dengan catu daya 12V untuk digunakan mencatu *electric fan* 12V DC. Sedangkan, catu daya 5V untuk digunakan mencatu rangkaian sensor suhu dan rangkaian *driver motor*.
2. Catu daya Arduino Mega 2560 didapat dari laptop/komputer dengan menghubungkan *port* komunikasi antara arduino dengan komputer menggunakan kabel USB dengan kecepatan transfer data sebesar 9600 *bit per second* (bps).
3. Nilai suhu mesin menggunakan celcius yang diukur dengan sensor suhu
4. Sensor suhu dihubungkan dengan modul rangkaian penguat karena tegangan keluaran sensor sangat kecil sehingga harus diperkuat agar dapat dibaca oleh arduino dengan rentang 0V hingga 5V. Modul ini adalah Max 6675

5. Sinyal keluaran dari modul rangkaian penguat ini digunakan sebagai masukan arduino yang kemudian diproses menggunakan kontroler PID.
6. Keluaran arduino berupa sinyal *pulse width modulation* (PWM) diberikan kepada rangkaian *driver motor*. Rangkaian ini memiliki fungsi untuk menguatkan sinyal PWM arduino dengan tegangan sebesar 0V hingga 5V menjadi 0V hingga 12V.
7. Terdapat dua buah *electric fan* DC 12V yang digerakkan oleh driver motor dengan tegangan 0V hingga 12 V
8. Kedua buah *electric fan* akan menyala apabila suhu mesin mobil melebihi 90°C dan akan padam apabila suhu telah mencapai 90°C
9. Sensor suhu akan terus mendeteksi nilai suhu selama mesin mobil dalam keadaan menyala

3.4.3 Rangkaian Catu Daya

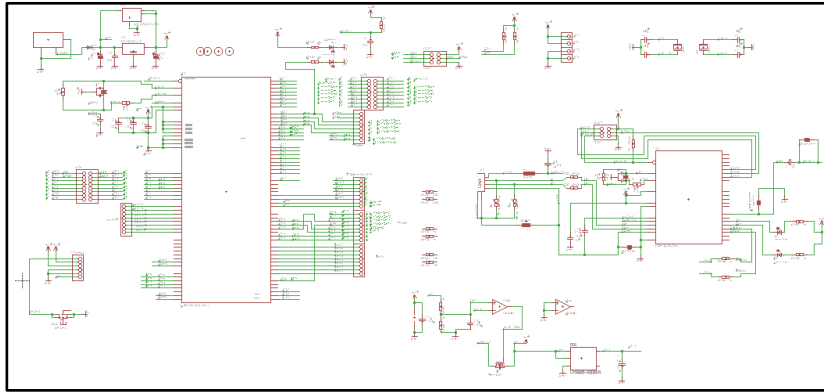
Rangkaian catu daya yang digunakan pada sistem ini adalah aki mobil yang memiliki keluaran sebesar 12V. catu daya DC 12V untuk *electric fan* diambil dari aki mobil dengan arus sebesar 55 A.



Gambar 3.9 Accu Motolite

3.4.4 Konfigurasi Pin Arduino Mega 2560

Pada alat ini digunakan mikrokontroler Arduino Mega 2560 sebagai pusat utama dalam proses pengendalian sistem. Konfigurasi I/O ditunjukkan dalam Gambar 3.10.



Gambar 3.10 Skematik Konfigurasi I/O Arduino Mega 2560.
Sumber: datasheet Arduino Mega 2560

Tabel masing-masing kegunaan pin Arduino Mega 2560 dalam sistem ini dapat ditunjukkan dalam tabel 3.4.

Tabel 3.4 Konfigurasi Pin Arduino Mega 2560 dalam Sistem.

No.	Pin	Fungsi
1.	Vin	Sebagai tegangan masukan untuk mencatu daya alat
2.	Gnd	Sebagai <i>ground</i> untuk mencatu daya alat
3.	2	Sebagai masukan LCD
4.	3	Sebagai tegangan keluaran untuk mencatu daya alat
5.	4	Sebagai masukan LCD
6.	5	Sebagai masukan LCD
7.	6	Sebagai masukan thermoCLK driver sensor suhu
8.	7	Sebagai masukan thermoCS driver sensor suhu
9.	8	Sebagai masukan thermoDO driver sensor suhu
10.	9	Sebagai masukan digital R_PWM <i>Electric Fan</i>
11.	10	Sebagai masukan digital L_PWM <i>Electric fan</i>
12.	11	Sebagai masukan LCD
13.	12	Sebagai masukan LCD
14.	13	Sebagai masukan LCD
15.	22	Sebagai masukan digital L_EN <i>Electric Fan</i>
16.	24	Sebagai masukan digital R_EN <i>Electric Fan</i>
17.	50	Sebagai masukan Miso Memory Card
18.	51	Sebagai masukan Mosi <i>memory card</i>
19.	52	Sebagai masukan sck <i>memory card</i>
20.	53	Sebagai masukan cs <i>memory card</i>

3.4.5 Konfigurasi Pin *Driver Motor*

Rangkaian *driver motor* digunakan untuk mengendalikan *electric fan*. Rangkaian *driver motor* yang digunakan adalah *driver motor H-Bridge* BTS7960 30A. Mode pengontrolan yang digunakan adalah mode sinyal *pulse width modulation* (PWM) dan dicatu dengan catu daya eksternal berupa aki mobil DC 12V. Tabel konfigurasi pin dari *driver motor* tersebut dapat dilihat dalam Tabel 3.5.

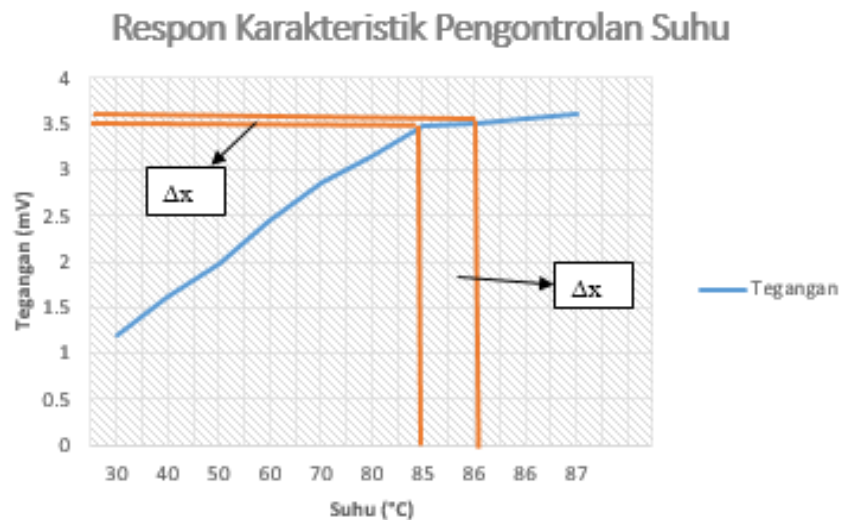
Tabel 3.5 Konfigurasi pin *Driver Motor H-Bridge* BTS7960 30A dalam Sistem.

No.	Pin	Fungsi
1.	Vin	Sebagai tegangan masukan untuk mencatu daya alat
2.	Gnd	Sebagai ground untuk mencatu daya alat
3.	D13	Sebagai R_PWM <i>electric fan</i>
4.	D12	Sebagai L_PWM <i>electric fan</i>

3.5 Perancangan Algoritma

3.5.1 Penentuan Karakteristik Pengontrolan Suhu

Untuk menemukan fungsi alih plant agar mendapatkan karakteristik *static gain* keadaan mantap dengan cara diberikan unit step masukan berupa pemasangan sensor didalam selang radiator bagian atas, kemudian mesin dinyalakan sampai mencapai setpoint yang diinginkan dan kemudian dilihat bagaimana keadaan mantapnya. Untuk menemukan parameter kontroler PID digunakan metode *ziegler-nichols* orde 1, dimana seperti pada Gambar 3.11 berikut ini.



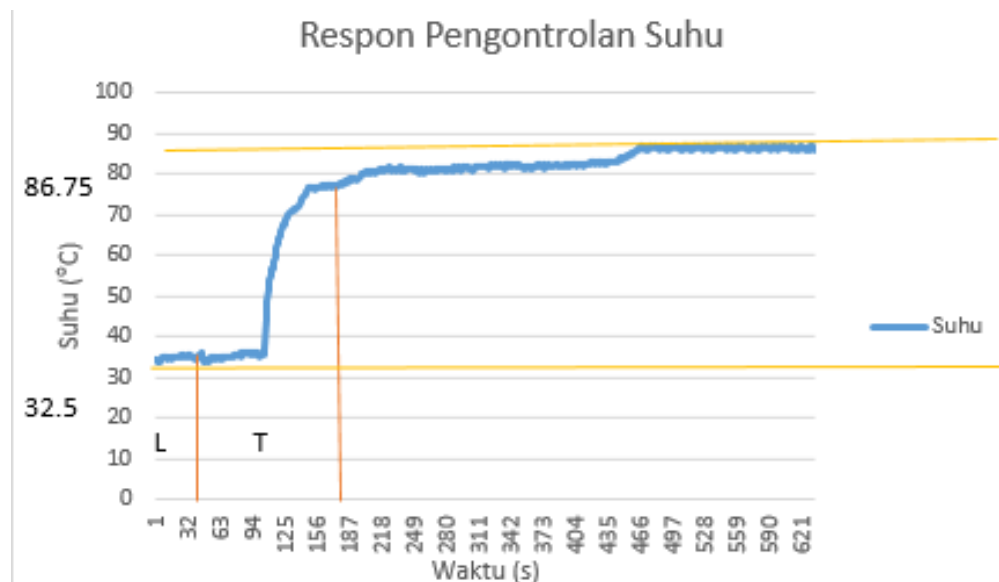
Gambar 3.11 Respon Karakteristik Pengontrolan Suhu.
Sumber: Perancangan

$$k = \frac{\Delta y}{\Delta u} = \frac{3.516 - 3.474}{86 - 85} = 0.042 \quad (3.4)$$

3.5.2 Parameter Kontroler PID dengan Metode Ziegler-Nichols

Dalam memenuhi tujuan performansi *loop* yang diinginkan, maka perlu ditambahkan sebuah kontroler di dalam sistem. Sedangkan, di dalam pengujian ini digunakan kontroler *proporsional*, *integral*, dan *differensial* (PID). Kontroler ini dibutuhkan dikarenakan karakteristik *plant* membutuhkan kontroler yang cepat dengan meminimalkan *overshoot*.

Langkah awal *tuning* parameter dengan menggunakan metode *ziegler-nichols* orde 1 adalah dengan menarik garis *tangent* pada titik infleksi grafik karakteristik *plant*. Kemudian mencari perpotongan garis *tangent* dengan garis nilai akhir, sehingga diperoleh nilai L (Keterlambatan transportasi proses) dan T (Konstanta waktu). Nilai L dan T tersebut nantinya akan digunakan untuk mencari parameter Kp, Ki, dan Kd



Gambar 3.12 Metode Ziegler-Nichols Orde 1 pada *Plant*.
Sumber: Perancangan

Dari pengujian tersebut dapat dicari nilai parameter L dan juga T. dalam grafik dapat diketahui bahwa nilai suhu awal adalah 32.5 dan nilai akhir adalah 97. Untuk mencari nilai T dapat dilakukan dalam perhitungan sebagai berikut:

$$\Delta \text{Suhu} = 86.75 - 32.5 = 54.25$$

Sedangkan, nilai T ditentukan dari 63.2% nilai suhu yang didapatkan dari perhitungan sebagai berikut:

$$\begin{aligned}
 63.2\% \Delta \text{Suhu} &= 32.5 + (63.2\%(86.75-32.5)) \\
 &= 32.5 + 34.286 \\
 &= 66.786
 \end{aligned}$$

Nilai T pada suhu 73.264 = 297 detik

Sehingga, pada saat 63.2% Δ Suhu, nilai T= 145 – 98 = 47 detik

L = 98 detik dan T = 47

Untuk melakukan *tuning* parameter dari kontroler PID menggunakan rumus yang ada pada Tabel 3.6

Tabel 3.6 Aturan Penalaan *Ziegler-Nichols* Berdasarkan Respon Unit Step Dari *Plant*

Tipe Kontroler	K_p	T_i	T_d
P	$\frac{T}{L}$	∞	0
PI	$0,9 \frac{T}{L}$	$\frac{L}{0,3}$	0
PID	$1,2 \frac{T}{L}$	$2 L$	$0,5 L$

Berdasarkan pengujian kurva respon step berbentuk s diatas. Sehingga dapat diperoleh persamaan matematis fungsi alih sistem adalah sebagai berikut:

$$\begin{aligned}
 Y(s) &= \frac{k}{Ts+1} \cdot e^{-sL} \\
 &= \frac{0.04}{47s+1} \cdot e^{-20s}
 \end{aligned} \tag{3.5}$$

Setelah mendapatkan nilai fungsi alih sistem, kemudian dengan menggunakan rumus yang ada pada Tabel 3.6 maka akan didapatkan nilai parameter K_p , K_i dan K_d adalah sebagai berikut:

$$K_p = 1.2 \frac{T}{L} = 1.2 \times \frac{47}{98} = 0.51 \quad (3.6)$$

$$T_i = 2L = 2 \times 98 = 196 \quad (3.7)$$

$$T_d = 0.5L = 0.5 \times 217 = 108.5 \quad (3.8)$$

$$K_i = \frac{K_p}{T_i} = \frac{0.51}{196} = 0.002 \quad (3.9)$$

$$K_p = \frac{K_d}{T_d} \longrightarrow K_d = K_p \times T_d = 0.51 \times 49 = 24.99 \quad (3.10)$$

Dari perhitungan diatas didapatkan nilai parameter $K_p = 0.51$, $K_i = 0.002$, dan $K_d = 24.99$. Sehingga akan mendapatkan persamaan transformasi *Laplace* berdasarkan dengan nilai K_p , K_i , dan K_d adalah sebagai berikut:

$$M(s) = [K_p + \frac{K_i}{s} + K_d s] E(s) \quad (3.11)$$

$$M(s) = [K_p s + K_i + K_d s^2] E(s) \quad (3.12)$$

3.5.3 Diskritisasi

Persamaan diatas belum dapat dimasukkan ke dalam mikrokontroler oleh karena itu persamaan kontinyu tersebut harus diubah ke dalam bentuk diskrit melalui Transformasi Z. Dalam transformasi Z dibutuhkan waktu *sampling* (T_s). Pada sistem ini menggunakan metode *Backward Difference* dikarenakan metode ini memiliki sensitifitas yang lebih rendah, sehingga sistem tidak akan mengalami perubahan atau osilasi saat diberi gangguan dan akan tetap mempertahankan performansinya. Dengan menggunakan metode *Backward Difference*, yaitu dengan mengganti s dalam persamaan analog bentuk s persamaan. Persamaan s tersebut setara dengan bentuk *Laplace*:

$$s = \frac{1 - z^{-1}}{T_s} \quad (3.12)$$

Maka persamaan 3.12 disubstitusikan ke dalam persamaan 3.11 akan menjadi:

$$\begin{aligned}
 M(z) &= [K_p + \frac{K_i}{\frac{1-z^{-1}}{T_s}} + K_d(\frac{1-z^{-1}}{T_s})]E(z) \\
 &= [K_p + \frac{K_i.T_s}{1-z^{-1}} + \frac{K_d}{T_s}(1-z^{-1})]E(z) \quad (3.13)
 \end{aligned}$$

Kemudian didapatkan rumus masing-masing parameter kontroler PID, adalah sebagai berikut:

$$\text{Kontroler Proporsional: } C_p(z) = K_p.E(z) \quad (3.14)$$

$$\begin{aligned}
 \text{Kontroler Integral: } C_i(z) &= \frac{K_i.T_s}{1-z^{-1}}.E(z) \\
 &= [C_i(z)z^{-1} + K_i.T_s.E(z)] \quad (3.15)
 \end{aligned}$$

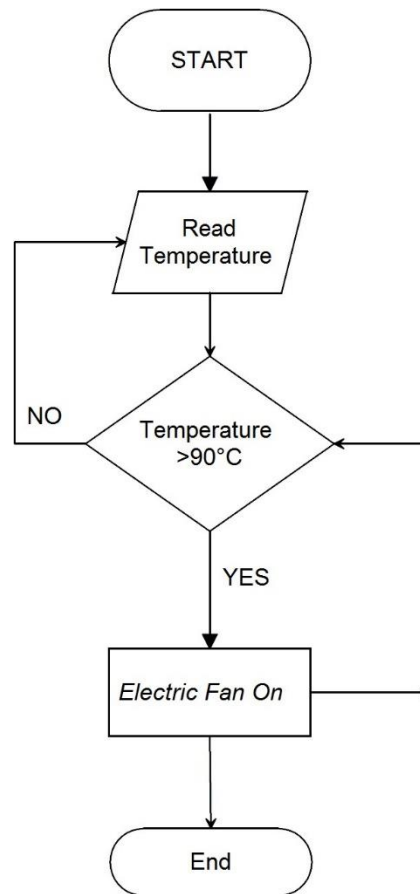
$$\begin{aligned}
 \text{Kontroler Diferensial: } C_d(z) &= \frac{K_d}{T_s}(1-z^{-1}).E(z) \\
 &= \frac{K_d}{T_s}[E(z) - E(z)z^{-1}] \quad (3.16)
 \end{aligned}$$

Persamaan 3.14 hingga persamaan 3.16 kemudian diubah ke dalam persamaan beda sehingga akan didapatkan persamaan 3.17 seperti di bawah ini:

$$M(k) = K_p.E(k) + (C_i(k-1) + K_i.T_s E(k)) + (\frac{K_d}{T_s}(E(k) - E(k-1))) \quad (3.17)$$

Dimana (k-1) adalah kondisi sebelumnya. Persamaan diatas lalu dimasukkan ke dalam program mikrokontroler.

3.5.4 Flowchat Program



Gambar 3.13 Flowchart

Sumber: Perancangan

Keterangan :

Pada saat mesin menyala, sistem akan mulai membaca suhu dari mesin tersebut. Jika suhu mesin melebihi 90°C maka *electric fan* akan menyala dengan otomatis, setelah itu sistem akan membandingkan dengan suhu yang telah ditetapkan. Tetapi jika suhu mesin kurang dari 90°C maka sistem akan mengalami siklus membaca suhu mesin dan membandingkan dengan suhu yang telah ditetapkan. Sistem akan berakhir saat mesin mobil dimatikan.

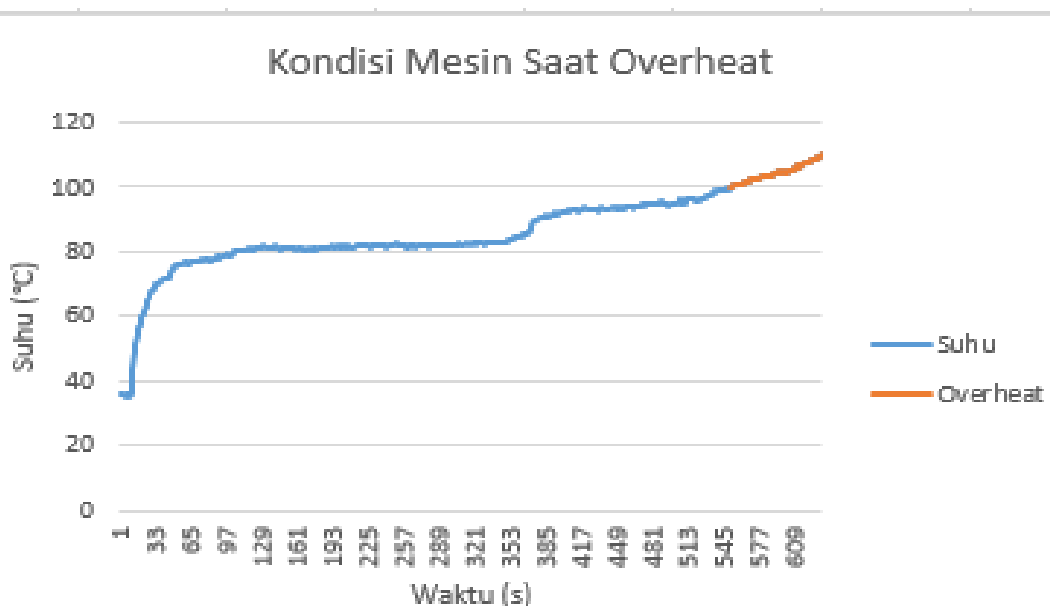
BAB IV

HASIL DAN PEMBAHASAN

Hasil dan pembahasan merupakan hasil dari pengujian keseluruhan sistem dengan parameter-parameter yang telah diberikan dari perhitungan. Parameter-parameter tersebut dilakukan pengujian apakah hasilnya sesuai dengan input yang sudah diberikan dan dapat diaplikasikan di dalam sistem yang telah dibuat. Selain itu juga untuk dapat melihat hasil respon yang dihasilkan. Pengujian keseluruhan sistem dibagi menjadi beberapa bagian yaitu:

- 4.1 Pengujian Suhu Mesin saat *Overheat*
- 4.2 Pengujian Penurunan Suhu Mesin Menggunakan Kontrol PID
- 4.3 Pengujian Penurunan Suhu Mesin Menggunakan Kontrol *On-Off*
- 4.4 Komparasi Kontrol PID dan *On-Off*

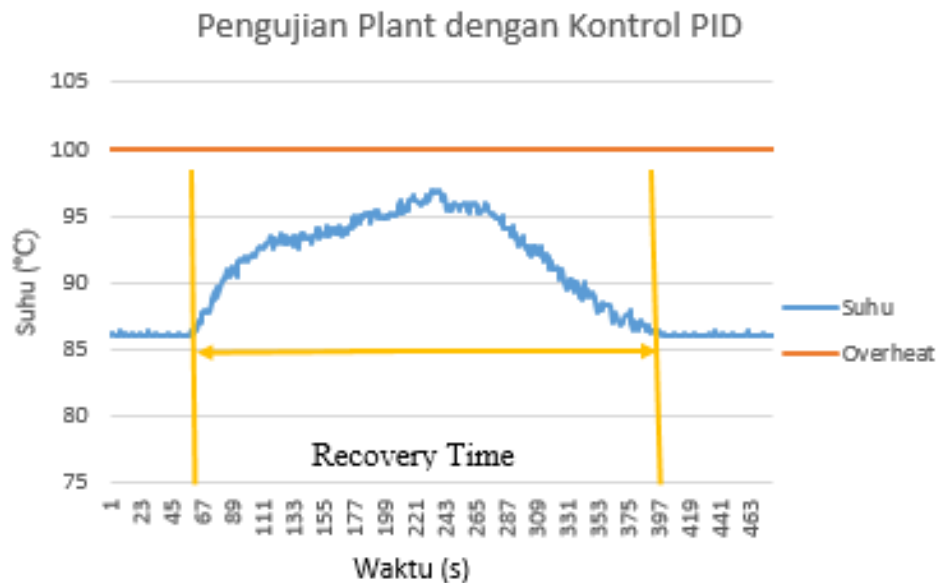
4.1 Pengujian Suhu Mesin saat *Overheat*



Gambar 4.1 Kondisi Suhu Mesin Saat Overheat
Sumber: Pengambilan Data Primer dan Sekunder

Suhu Optimal mesin mobil dalam penelitian ini adalah 86°C, mesin tetap dapat bekerja hingga mencapai suhu 100°C. Apabila suhu mesin melebihi 100°C maka mesin berpotensi akan mengalami kerusakan akibat *Overheat*. Kerusakan yang dapat terjadi adalah memuainya material-material yang terdapat pada mesin mobil sehingga membutuhkan perbaikan.

4.2 Pengujian Penurunan Suhu Mesin Menggunakan Kontrol PID

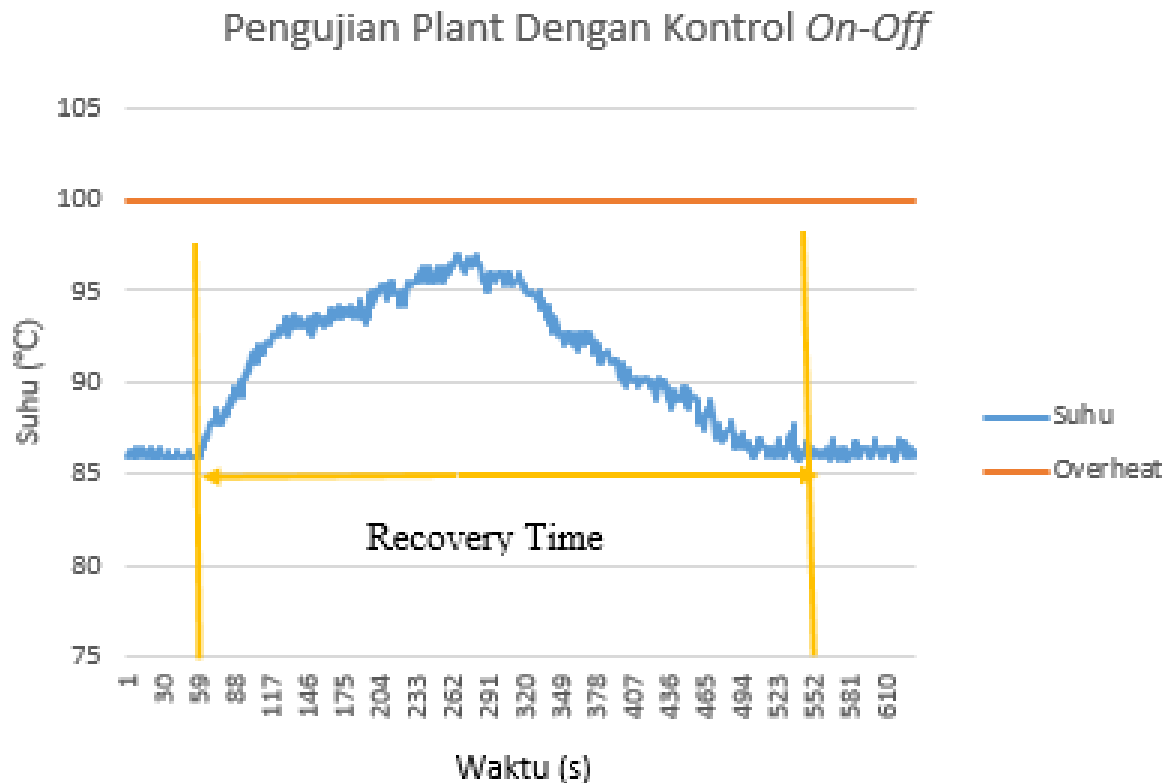


Gambar 4.2 Pengujian Plant Dengan Kontrol PID

Sumber: Perancangan

Pada gambar 4.2 dapat dilihat bahwa suhu ideal mesin mobil pada kisaran 86°C. Kemudian untuk melihat respon PID dilakukan dengan menekan pedal gas pada 3000 rpm selama 5 menit. Gangguan terjadi pada detik ke 63 hingga detik ke 405 dengan nilai kenaikan suhu sebesar 11°C, namun setelah detik ke 406 nilai suhu kembali mengikuti *setting point* dengan *error steady state* sebesar 2% dan *recovery time* sebesar 342 detik. *Electric Fan* pada sistem ini didesain agar menyala pada suhu 90°C dan padam apabila mencapai suhu 86°C. Pada penelitian ini respon dari plant control PID mampu menjaga suhu dibawah ambang batas *Overheat*.

4.3 Pengujian Penurunan Suhu Mesin Menggunakan Kontrol *On-Off*



Gambar 4.3 Pengujian Plant Kontrol *On-Off*
Sumber: Perancangan

Pada gambar 4.3 dapat dilihat bahwa suhu ideal mesin mobil pada kisaran 86°C. Kemudian untuk melihat respon Kontroler *On-Off* dilakukan dengan menekan pedal gas pada 3000rpm selama 5 menit. Gangguan terjadi pada detik ke 60 hingga detik ke 568 dengan nilai kenaikan suhu sebesar 11°C, namun setelah detik ke 569 nilai suhu kembali mengikuti *setting point* dengan *error steady state* sebesar 3% dan *recovery time* sebesar 509 detik. *Electric Fan* pada sistem ini didesain agar menyala pada suhu 90°C dan padam apabila mencapai suhu 86°C *Electric Fan* akan padam. Pada penelitian ini respon dari plant control *On-Off* mampu menjaga suhu dibawah ambang batas *Overheat*.

4.4 Komparasi Kontrol PID dan *On-Off*

Tabel 4.1 Komparasi Kontrol PID dan Kontrol *On-Off*

	Suhu setpoint	Suhu Akhir	Awal Kenaikan Suhu	<i>Steady</i>	<i>Electric Fan On</i>	Recovery Time
PID	86°C	97°C	63s	406s	90°C	342s
<i>On-Off</i>	86°C	97°C	60s	569s	90°C	509s

Dari tabel 4.1 dapat dilihat bahwa control PID dan *On-Off* mempunyai Setpoint yang sama yaitu 86°C, kemudian suhu saat *Electric Fan* menyala juga sama yaitu pada suhu 90°C. Kenaikan suhu pada Kontrol PID dimulai pada detik ke 63, mengalami kenaikan suhu sebesar 11°C yaitu pada 97°C. Pada suhu 90°C *Electric Fan* menyala, akan tetapi suhu terus naik sampai pada 97°C. Pada detik ke 406s Kontrol PID telah menurunkan suhu kembali ke *Setpoint* awal dengan *Recovery Time* sebesar 342s.

Pada Kontrol *On-Off* kenaikan suhu dimulai pada detik ke 60 mengalami kenaikan suhu sebesar 11°C yaitu pada 97°C. Pada suhu 90°C *Electric Fan* menyala, akan tetapi suhu terus naik sampai pada 97°C. Pada detik ke 569 Kontrol *On-Off* telah menurunkan suhu kembali ke *Setpoint* awal dengan *Recovery Time* sebesar 509s.

Dari tabel 4.1 dapat disimpulkan bahwa kontrol PID mempunyai respon yang lebih baik daripada kontrol *On-Off*. Hal ini dapat dilihat dari selisih *Recovery Time* kedua kontroler yaitu sebesar 167s.

BAB V

PENUTUP

5.1 Kesimpulan

Berdasarkan pada perancangan alat dan hasil pengujian dapat diambil beberapa kesimpulan, yaitu:

1. Hasil pengujian dari respon sistem PID pada pengontrolan temperatur dengan Suhu sebagai indikator, didapatkan nilai $L = 98$ detik, $T = 47$ detik, dan $K = 0.04$.
2. Hasil pengujian untuk menentukan parameter PID dari sistem pengontrolan ini adalah sebagai berikut:
 - a. Respon sistem PID pada *setting point* tetap 86°C dari keadaan awal 32.5 ; respon sistem telah menuju ke titik kestabilan untuk nilai parameter PID $K_p = 0.51$, $T_i = 196$, dan $T_d = 49$.
 - b. Pada pengujian pengontrolan suhu dengan temperatur sebagai indikator dengan gangguan menekan pedal gas pada 3000 rpm selama 5 menit sudah dapat mencapai kestabilan dengan *error steady state* sebesar 2% dan *recovery time* sebesar 342 detik.
3. Pada Pengujian pengontrolan suhu menggunakan kontroler *on-off* dengan temperatur sebagai indikator dengan gangguan menekan pedal gas pada 3000 rpm selama 5 menit sudah dapat mencapai kestabilan dengan *error steady state* sebesar 3% dan *recovery time* sebesar 509 detik. Dapat disimpulkan bahwa penggunaan kontrol PID mempunyai hasil yang lebih baik dibanding menggunakan kontrol *On-Off*

5.2 Saran

Di dalam perancangan dan pembuatan alat dalam pengujian ini masih banyak terdapat kelemahan dan kekurangan. Oleh karena itu, terdapat beberapa hal yang dapat dijadikan saran untuk kesempurnaan alat lebih lanjut, sebagai berikut:

1. Untuk pengaplikasian pada kendaraan lain perlu penyesuaian *setpoint* dengan suhu kendaraan tersebut karena *setpoint* setiap mobil berbeda.
2. Sebaiknya dipasang saklar otomatis saat mesin mendekati batas maksimal *Overheat*, kemudian dapat dinyalakan kembali pada saat suhu sudah turun

DAFTAR PUSTAKA

- Badmianto, Trigas. 2006. Sistem Kontrol Analog Suhu Mesin Mobil dengan Op-Amp (Operational Amplifier). Malang: PPPPTK VEDC
- Bolton, W. 2004. Sistem Instrumentasi dan Sistem Kontrol. Jakarta: Penerbit Erlangga
- Daryanto. 2002. Pemeliharaan Sistem Pendingin dan Sistem Pelumasan Mobil. Bandung: Krama Widya
- Irfan, Ade S. 2007. Analisis Sistem Pendinginan Pada Mesin Isuzu Panther. Proyek Akhir. Semarang: Universitas Negeri Semarang.
- Kadir, Abdul. 2006. Transportasi: Peran dan Dampaknya Dalam Pertumbuhan Ekonomi Nasional. Medan: Universitas Sumatera Utara
- Ogata, K. 2010. Modern Control Engineering, Pearson Education, Inc., publishing as Prentice Hall, One Lake Street, Upper Saddle River, New Jersey 07458. Fifth edition.
- Peraturan Pemerintah Republik Indonesia Nomor 44 Tahun 1993 Tentang Kendaraan dan Pengemudi. 16 November 2016. <http://www.hubdat.dephub.go.id/peraturan-pemerintah/82-pp-no-44-tahun-1993-ttg-kendaraan-dan-pengemudi>
- Perkembangan Jumlah Kendaraan Bermotor Menurut Jenis Tahun 1949-2014. 11 November 2016. <http://www.bps.go.id/linkTableDinamis/view/id/1133>

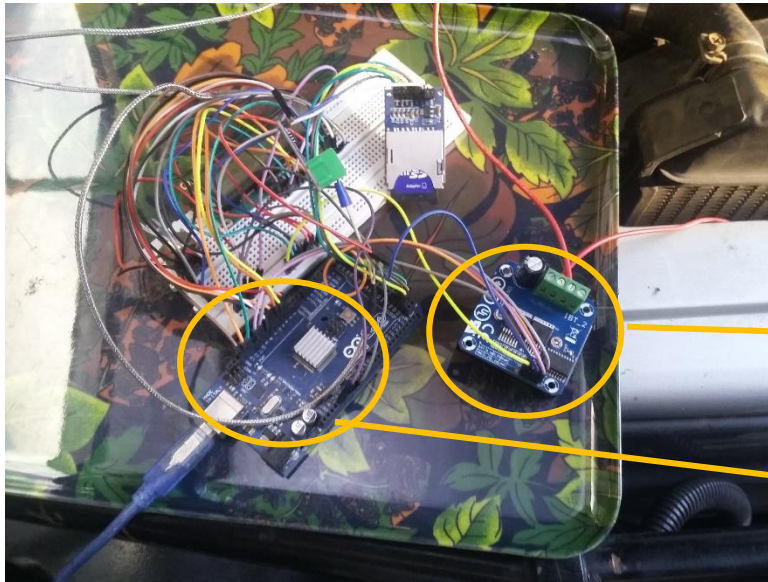
LAMPIRAN I

FOTO ALAT

FOTO ALAT



Pemasangan sensor
termokopel pada selang
radiator bagian atas



Driver Motor
BTS7960

Arduino Mega 2560



LAMPIRAN II

LISTING PROGRAM

```

#include <btsmotordriver.h>

#include <SimpleTimer.h>

#include "max6675.h"

#include <LiquidCrystal.h>

#include <Wire.h>

#include <SPI.h>

//#include <SD.h>

#include "SdFat.h"

SdFat SD;

SimpleTimer timer;


//LCD

//LiquidCrystal lcd(30, 35, 31, 32, 33, 34);

LiquidCrystal lcd(12, 11, 5, 4, 13, 2);


// the setup routine runs once when you press reset:

uint8_t block[8] = {0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF};


//sensor temperatur


int thermoDO = 8;

int thermoCS = 7;

int thermoCLK = 6;


MAX6675 thermocouple(thermoCLK, thermoCS, thermoDO);

int vccPin = 3;

int gndPin = 2;

```

```
double setPoint, error, dError, sError, lError, dTime, PID, kP, kI, kD, Ts, KP_PID, KI_PID, KD_PID,
KI_PID_LAST, error1;
```

```
long pwm;
```

```
long delayCampur;
```

```
long delayPompa;
```

```
//Driver Motor
```

```
//BTS7960MotorDriver objectname(R_EN,L_EN,RPWM,LPWM);
```

```
BTS7960MotorDriver fanmotor(24,22,9,10);
```

```
//datalogger sdcard
```

```
const int batteryPin = A7;    // battery voltage is on pin A7
```

```
const int chipSelect = 53;    // SPI chip select for SD card
```

```
const int cardDetect = 26;    // pin that detects whether the card is there
```

```
const int writeLed = 28;      // LED indicator for writing to card
```

```
const int errorLed = 33;      // LED indicator for error
```

```
long lastWriteTime = 0;       // timestamp for last write attempt
```

```
long interval = 1000;         // time between readings
```

```
char fileName[] = "datalog15.csv"; // filename to save on SD card
```

```
float temp;
```

```
void setup()
```

```
{
```

```
    Serial.begin (9600);
```

```
//LCD
```

```
lcd.begin(16, 2);  
//lcd.createChar(0, degree);  
lcd.createChar(2, block);  
lcd.setCursor(0, 0);  
lcd.print("MOTOR:");  
//lcd.print(thermocouple.readCelsius());  
lcd.setCursor(1, 1);  
lcd.print("C");  
lcd.setCursor(14, 1);  
lcd.print("H");  
  
///  
//datalogger sdcard  
// initialize LED and cardDetect pins:  
pinMode(writeLed, OUTPUT);  
pinMode(errorLed, OUTPUT);  
pinMode(cardDetect, INPUT_PULLUP);  
  
// startSDCard() blocks everything until the card is present  
// and writable:  
if (startSDCard() == true) {  
    Serial.println("card initialized.");  
    delay(100);  
    // open the log file:  
    File logFile = SD.open(fileName, FILE_WRITE);  
    // write header columns to file:  
    if (logFile) {  
        logFile.println("Suhu Celcius,Suhu Fahrenheit");  
        logFile.close();  
    }
```



```
    }  
  }  
  /**/  
  
  // use Arduino pins temperature  
  pinMode(vccPin, OUTPUT); digitalWrite(vccPin, HIGH);  
  pinMode(gndPin, OUTPUT); digitalWrite(gndPin, LOW);  
  
  Serial.println("MAX6675 test");  
  //wait for MAX chip to stabilize  
  delay(500);  
  
  temp = 0;  
  error = 0;  
  dError = 0;  
  sError = 0;  
  lError = 0;  
  dTime = 0;  
  PID = 0;  
  
  setPoint = 86;  
  kP = 1.89;  
  kI = 0.002;  
  kD = 24.99;  
  Ts = 114;
```

```
    timer.setInterval(500, readtempTimer);  
timer.setInterval(1000, driverEnableTimer);  
timer.setInterval(1000, kondisitempTimer);  
// timer.setInterval(1000, computeTimer);  
timer.setInterval(1000, printdatalcdTimer);  
timer.setInterval(1000, LcdgraphicTimer);  
timer.setInterval(1000, dataloggerTimer);
```

```
}
```

```
void readtempTimer()
```

```
{
```

```
    readtemp();
```

```
}
```

```
void driverEnableTimer()
```

```
{
```

```
    driverEnable();
```

```
}
```

```
void kondisitempTimer()
```

```
{
```

```
    kondisitemp();
```

```
}
```

```
void computeTimer()
```

```
{
```

```
    compute();
```

```
}
```

```
void LcdgraphicTimer()
```

```
{
```

```
Lcdgraphic();
}
void printdatalcdTimer()
{
    printdatalcd();
}
void dataloggerTimer()
{
    datalogger();
}

void loop()
{
    timer.run();

}
void readtemp()
{
    // basic readout test, just print the current temp

    Serial.print("C = ");
    Serial.println(thermocouple.readCelsius());
    //Serial.print("F = ");
    //Serial.println(thermocouple.readFahrenheit());

    delay(1000);
}
void driverEnable()
```

```

{
// digitalWrite(L_EN_fan, HIGH);
// digitalWrite(R_EN_fan, OUTPUT);
// digitalWrite(LPWM_fan, LOW);
fanmotor.Ready();
}

void kondisitemp()
{
if (thermocouple.readCelsius() >= 25 && thermocouple.readCelsius() <= 80)
{
// analogWrite(RPWM_fan,0);
// analogWrite(LPWM_fan,0);
// fanmotor.Ready();
// fanmotor.Ready();
fanmotor.Stop();
//Serial.println("FAN OFF");
lcd.setCursor(8, 0);
lcd.print("FAN OFF");
}

else if (thermocouple.readCelsius() > 90 && thermocouple.readCelsius() <= 100)
{
// analogWrite(RPWM_fan,150);
// analogWrite(LPWM_fan,0);
//fanmotor.Ready();
fanmotor.TurnRight(255);
//Serial.println("FAN ON");
lcd.setCursor(8, 0);
lcd.print("FAN ON");
}
}

```

```

    else if (thermocouple.readCelsius() > 105 && thermocouple.readCelsius() < 110)
    {
//  analogWrite(RPWM_fan,255);
//  analogWrite(LPWM_fan,0);
//fanmotor.Ready();
fanmotor.TurnRight(255);
//Serial.println("FAN ON");
    lcd.setCursor(8, 0);
    lcd.print("FAN ON");
    }
}
void compute()
{
    error = setPoint-temp;

    KP_PID = kP*error;
    KI_PID = KI_PID_LAST + (kI*Ts*error);
    KD_PID = ((kD/Ts)*error)-((kD/Ts)*error1);
    PID = KP_PID + KI_PID + KD_PID;

    KI_PID_LAST = KI_PID;
    error1 = error;

    if(PID>255)
        pwm = 255;
    else if(PID<-255)
        pwm = 255;
    else
    {

```

```

    if(PID>0)
        pwm = (int) PID;
    else
        pwm = (int)(PID*(-1));
    }
}

void printdatalcd()
{

}

void Lcdgraphic(){
    if (thermocouple.readCelsius() >= 35) {
        lcd.setCursor(3,1);
        lcd.write(2);

    } else {
        lcd.setCursor(3,1);
        lcd.print(" ");

    }

    if (thermocouple.readCelsius() >= 45) {
        lcd.setCursor(4,1);
        lcd.write(2);

    } else {
        lcd.setCursor(4,1);
        lcd.print(" ");

    }
}

```

```
if (thermocouple.readCelsius() >= 55) {  
    lcd.setCursor(5,1);  
    lcd.write(2);  
} else {  
    lcd.setCursor(5,1);  
    lcd.print(" ");  
  
}  
  
if (thermocouple.readCelsius() >= 65) {  
    lcd.setCursor(6,1);  
    lcd.write(2);  
} else {  
    lcd.setCursor(6,1);  
    lcd.print(" ");  
  
}  
  
if (thermocouple.readCelsius() >= 75) {  
    lcd.setCursor(7,1);  
    lcd.write(2);  
} else {  
    lcd.setCursor(7,1);  
    lcd.print(" ");  
  
}  
  
if (thermocouple.readCelsius() >= 85) {  
    lcd.setCursor(8,1);  
    lcd.write(2);  
} else {  
    lcd.setCursor(8,1);
```

```
    lcd.print(" ");

}

if (thermocouple.readCelsius() >= 95) {
    lcd.setCursor(9,1);
    lcd.write(2);
} else {
    lcd.setCursor(9,1);
    lcd.print(" ");

}

if (thermocouple.readCelsius() >= 105) {
    lcd.setCursor(10,1);
    lcd.write(2);
} else {
    lcd.setCursor(10,1);
    lcd.print(" ");

}

if (thermocouple.readCelsius() >= 115) {
    lcd.setCursor(11,1);
    lcd.write(2);
} else {
    lcd.setCursor(11,1);
    lcd.print(" ");

}

if (thermocouple.readCelsius() >= 125) {
    lcd.setCursor(12,1);
```



```

    lcd.write(2);

} else {
    lcd.setCursor(12,1);
    lcd.print(" ");
}

}}

void datalogger()
{
    // if the cart's not there, don't do anything more:
    if (digitalRead(cardDetect) == LOW) {
        digitalWrite(errorLed, HIGH);
        return;
    }
    // turn of the error LED:
    digitalWrite(errorLed, LOW);

    // read sensors every 10 seconds
    if (millis() - lastWriteTime >= interval) {
        File logFile = SD.open(fileName, FILE_WRITE); // open the log file
        if (logFile) { // if you can write to the log file,
            digitalWrite(writeLed, HIGH); // turn on the write LED

            logFile.print(thermocouple.readCelsius()); // print battery voltage to the log
            logFile.print(","); // print comma to the log
            logFile.println(thermocouple.readFahrenheit()); // print sensor voltage & newline to the log
            logFile.close(); // close the file
        }
    }
}

```

```
Serial.print(thermocouple.readCelsius());  
  
Serial.print(",");  
  
Serial.println(thermocouple.readFahrenheit());
```

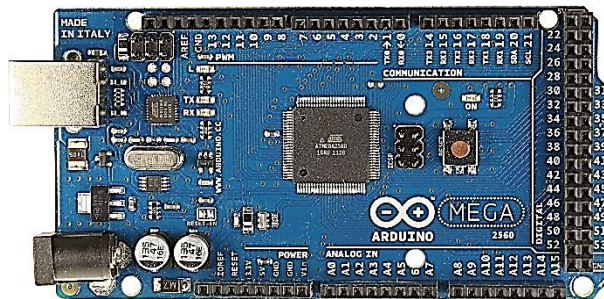
```
    // update the last attempted save time:  
    lastWriteTime = millis();  
}  
digitalWrite(writeLed, LOW);    // turn off the write LED  
}  
}
```

```
boolean startSDCard() {  
    // Wait until the card is inserted:  
    while (digitalRead(cardDetect) == LOW) {  
        Serial.println("Waiting for card...");  
        digitalWrite(errorLed, HIGH);  
        delay(750);  
    }  
  
    // wait until the card initialized successfully:  
    while (!ISD.begin(chipSelect)) {  
        digitalWrite(errorLed, HIGH); // turn on error LED  
        Serial.println("Card failed");  
        delay(750);  
    }  
    return true;  
}
```


LAMPIRAN III

DATASHEET

Arduino Mega 2560



Technical specs

Microcontroller	ATmega2560
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limit)	6-20V
Digital I/O Pins	54 (of which 15 provide PWM output)
Analog Input Pins	16
DC Current per I/O Pin	20 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	256 KB of which 8 KB used by bootloader
SRAM	8 KB
EEPROM	4 KB
Clock Speed	16 MHz
Length	101.52 mm
Width	53.3 mm
Weight	37 g

Beberapa pin memiliki fungsi khusus:

- *Serial*: 0 (RX) dan 1 (TX); *Serial 1*: 19 (RX) dan 18 (TX); *Serial 2*: 17 (RX) dan 16 (TX); *Serial 3*: 15 (RX) dan 14 (TX). Digunakan untuk menerima (RX) dan mengirimkan (TX) TTL data serial. Pin ini dihubungkan ke pin yang berkaitan dengan chip Serial ATmega8U2 USB-to-TTL.
- *Eksternal interrupts*: 2 (*interrupt 0*), 3 (*interrupt 1*), 18 (*interrupt 5*), 19 (*interrupt 4*), 20 (*interrupt 3*), dan 2 (*interrupt 2*). Pin ini dapat dikonfigurasi untuk memicu *interrupt* pada nilai yang rendah, dengan batasan tepi naik atau turun, atau perubahan nilai.
- *PWM*: 0 - 13. Menyediakan output PWM 8-bit dengan fungsi *analogWrite()*.
- *SPI*: 50 (MISO), 51 (MOSI), 52 (SCK), 53 (SS). Pin ini mendukung komunikasi SPI menggunakan *SPI library*.
- *LED*: 13. Ada *built-in* LED terhubung ke pin digital 13. Ketika pin bernilai nilai *high* LED menyala dan ketika pin bernilai *low* LED mati.
- *I²C*: 20 (SDA) dan 21 (SCL). Dukungan *I²C* (TWI) komunikasi menggunakan *wire*.



Cold-Junction-Compensated K-Thermocouple-to-Digital Converter (0°C to +1024°C)

MAX6675

General Description

The MAX6675 performs cold-junction compensation and digitizes the signal from a type-K thermocouple. The data is output in a 12-bit resolution, SPI™-compatible, read-only format.

This converter resolves temperatures to 0.25°C, allows readings as high as +1024°C, and exhibits thermocouple accuracy of 8LSBs for temperatures ranging from 0°C to +700°C.

The MAX6675 is available in a small, 8-pin SO package.

Features

- ◆ Direct Digital Conversion of Type -K Thermocouple Output
- ◆ Cold-Junction Compensation
- ◆ Simple SPI-Compatible Serial Interface
- ◆ 12-Bit, 0.25°C Resolution
- ◆ Open Thermocouple Detection

Ordering Information

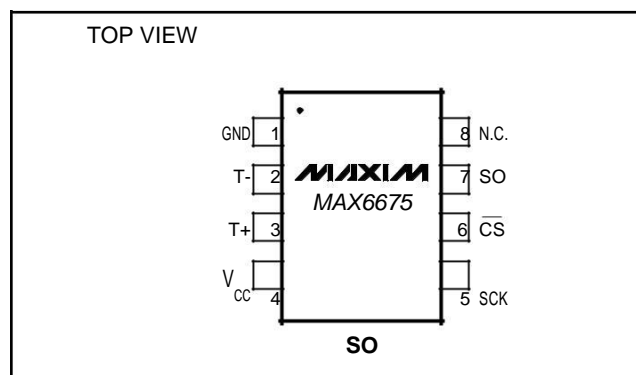
PART	TEMP RANGE	PIN-PACKAGE
MAX6675ISA	-20°C to +85°C	8 SO

Applications

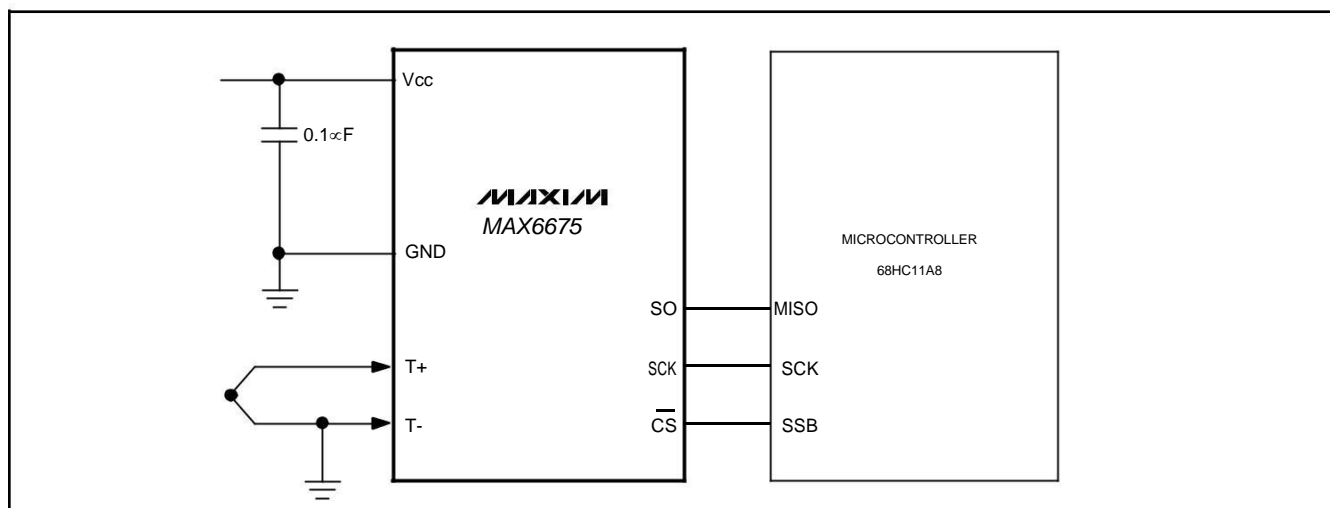
Industrial
Appliances
HVAC
Automotive

SPI is a trademark of Motorola, Inc.

Pin Configuration



Typical Application Circuit



Maxim Integrated Products 1

For pricing, delivery, and ordering information, please contact Maxim/Dallas Direct! at 1-888-629-4642, or visit Maxim's website at www.maxim-ic.com.

Cold-Junction-Compensated K-Thermocouple-to-Digital Converter (0°C to +1024°C)

ABSOLUTE MAXIMUM RATINGS

Supply Voltage (VCC to GND)	-0.3V to +6V	Storage Temperature Range	-65°C to +150°C
SO, SCK, CS, T-, T+ to GND	-0.3V to VCC + 0.3V	Junction Temperature	+150°C
SO Current	50mA	SO Package	
ESD Protection (Human Body Model)	±2000V	Vapor Phase (60s)	+215°C
Continuous Power Dissipation (TA = +70°C)		Infrared (15s)	+220°C
8-Pin SO (derate 5.88mW/°C above +70°C)	471mW	Lead Temperature (soldering, 10s)	+300°C
Operating Temperature Range	-20°C to +85°C		

Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those indicated in the operational sections of the specifications is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

ELECTRICAL CHARACTERISTICS

(VCC = +3.0V to +5.5V, TA = -20°C to +85°C, unless otherwise noted. Typical values specified at +25°C.) (Note 1)

PARAMETER	SYMBOL	CONDITIONS		MIN	TYP	MAX	UNITS
Temperature Error		THERMOCOUPLE = +700°C, TA = +25°C (Note 2)	VCC = +3.3V	-5		+5	LSB
			VCC = +5V	-6		+6	
		THERMOCOUPLE = 0°C to +700°C, TA = +25°C (Note 2)	VCC = +3.3V	-8		+8	
			VCC = +5V	-9		+9	
		THERMOCOUPLE = +700°C to +1000°C, TA = +25°C (Note 2)	VCC = +3.3V	-17		+17	
			VCC = +5V	-19		+19	
Thermocouple Conversion Constant					10.25		µV/LSB
Cold-Junction Compensation Error		TA = -20°C to +85°C (Note 2)	VCC = +3.3V	-3.0		+3.0	°C
			VCC = +5V	-3.0		+3.0	
Resolution					0.25		°C
Thermocouple Input Impedance					60		kΩ
Supply Voltage	VCC			3.0		5.5	V
Supply Current	ICC				0.7	1.5	mA
Power-On Reset Threshold		VCC rising		1	2	2.5	V
Power-On Reset Hysteresis					50		mV
Conversion Time		(Note 2)			0.17	0.22	s
SERIAL INTERFACE							
Input Low Voltage	VIL					0.3 x VCC	V
Input High Voltage	VIH					0.7 x VCC	V
Input Leakage Current	ILEAK	VIN = GND or VCC				±5	µA
Input Capacitance	CIN					5	pF

Cold-Junction-Compensated K-Thermocouple-to-Digital Converter (0°C to +1024°C)

ELECTRICAL CHARACTERISTICS (continued)

(VCC = +3.0V to +5.5V, TA = -20°C to +85°C, unless otherwise noted. Typical values specified at +25°C.) (Note 1)

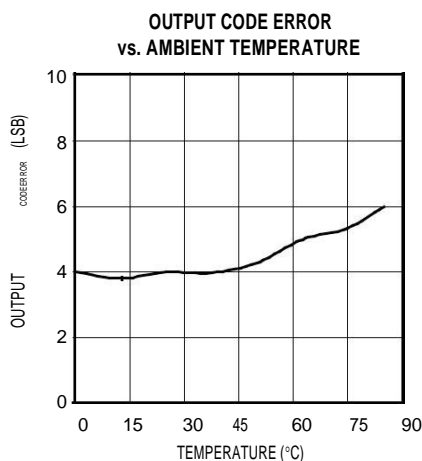
PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
Output High Voltage	VOH	ISOURCE = 1.6mA	VCC - 0.4			V
Output Low Voltage	VOL	ISINK = 1.6mA			0.4	V
TIMING						
Serial Clock Frequency	fSCL				4.3	MHz
SCK Pulse High Width	tCH		100			ns
SCK Pulse Low Width	tCL		100			ns
CSB Fall to SCK Rise	tcSS	CL = 10pF	100			ns
CSB Fall to Output Enable	tdV	CL = 10pF			100	ns
CSB Rise to Output Disable	tTR	CL = 10pF			100	ns
SCK Fall to Output Data Valid	tDO	CL = 10pF			100	ns

Note 1: All specifications are 100% tested at TA = +25°C. Specification limits over temperature (TA = TMIN to TMAX) are guaranteed by design and characterization, not production tested.

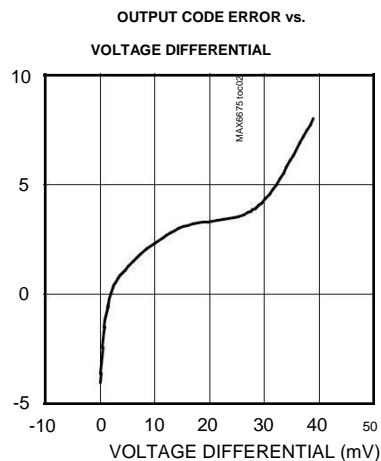
Note 2: Guaranteed by design. Not production tested.

Typical Operating Characteristics

(VCC = +3.3V, TA = +25°C, unless otherwise noted.)



MAX6675



Cold-Junction-Compensated K-Thermocouple-to-Digital Converter (0°C to +1024°C)

Pin Description

PIN	NAME	FUNCTION
1	GND	Ground
2	T-	Alumel Lead of Type-K Thermocouple. Should be connected to ground externally.
3	T+	Chromel Lead of Type-K Thermocouple
4	VCC	Positive Supply. Bypass with a 0.1µF capacitor to GND.
5	SCK	Serial Clock Input
6	CS	Chip Select. Set CS low to enable the serial interface.
7	SO	Serial Data Output
8	N.C.	No Connection

Detailed Description

The MAX6675 is a sophisticated thermocouple-to-digital converter with a built-in 12-bit analog-to-digital converter (ADC). The MAX6675 also contains cold-junction compensation sensing and correction, a digital controller, an SPI-compatible interface, and associated control logic.

The MAX6675 is designed to work in conjunction with an external microcontroller (µC) or other intelligence in thermostatic, process-control, or monitoring applications.

Temperature Conversion

The MAX6675 includes signal-conditioning hardware to convert the thermocouple's signal into a voltage compatible with the input channels of the ADC. The T+ and T- inputs connect to internal circuitry that reduces the introduction of noise errors from the thermocouple wires.

Before converting the thermoelectric voltages into equivalent temperature values, it is necessary to compensate for the difference between the thermocouple cold-junction side (MAX6675 ambient temperature) and a 0°C virtual reference. For a type-K thermocouple, the voltage changes by 41µV/°C, which approximates the thermocouple characteristic with the following linear equation:

$$V_{OUT} = (41\mu V / ^\circ C) \times (T_R - T_{AMB})$$

Where:

V_{OUT} is the thermocouple output voltage (µV).

T_R is the temperature of the remote thermocouple junction (°C).

T_{AMB} is the ambient temperature (°C).

Cold-Junction Compensation

The function of the thermocouple is to sense a difference in temperature between two ends of the thermocouple wires. The thermocouple's hot junction can be read from 0°C to +1023.75°C. The cold end (ambient temperature of the board on which the MAX6675 is mounted) can only range from -20°C to +85°C. While the temperature at the cold end fluctuates, the MAX6675 continues to accurately sense the temperature difference at the opposite end.

The MAX6675 senses and corrects for the changes in the ambient temperature with cold-junction compensation. The device converts the ambient temperature reading into a voltage using a temperature-sensing diode. To make the actual thermocouple temperature measurement, the MAX6675 measures the voltage from the thermocouple's output and from the sensing diode. The device's internal circuitry passes the diode's voltage (sensing ambient temperature) and thermocouple voltage (sensing remote temperature minus ambient temperature) to the conversion function stored in the ADC to calculate the thermocouple's hot-junction temperature.

Optimal performance from the MAX6675 is achieved when the thermocouple cold junction and the MAX6675 are at the same temperature. Avoid placing heat-generating devices or components near the MAX6675 because this may produce cold-junction-related errors.

Digitization

The ADC adds the cold-junction diode measurement with the amplified thermocouple voltage and reads out the 12-bit result onto the SO pin. A sequence of all zeros means the thermocouple reading is 0°C. A sequence of all ones means the thermocouple reading is +1023.75°C.

Cold-Junction-Compensated K-Thermocouple-to-Digital Converter (0°C to +1024°C)

MAX6675

Applications Information

Serial Interface

The *Typical Application Circuit* shows the MAX6675 interfaced with a microcontroller. In this example, the MAX6675 processes the reading from the thermocouple and transmits the data through a serial interface. Force CS low and apply a clock signal at SCK to read the results at SO. Forcing CS low immediately stops any conversion process. Initiate a new conversion process by forcing CS high.

Force CS low to output the first bit on the SO pin. A complete serial interface read requires 16 clock cycles. Read the 16 output bits on the falling edge of the clock. The first bit, D15, is a dummy sign bit and is always zero. Bits D14–D3 contain the converted temperature in the order of MSB to LSB. Bit D2 is normally low and goes high when the thermocouple input is open. D1 is low to provide a device ID for the MAX6675 and bit D0 is three-state.

Figure 1a is the serial interface protocol and Figure 1b shows the serial interface timing. Figure 2 is the SO output.

Open Thermocouple

Bit D2 is normally low and goes high if the thermocouple input is open. In order to allow the operation of the open thermocouple detector, T- must be grounded. Make the ground connection as close to the GND pin as possible.

Noise Considerations

The accuracy of the MAX6675 is susceptible to power-supply coupled noise. The effects of power-supply noise can be minimized by placing a 0.1µF ceramic bypass capacitor close to the supply pin of the device.

Thermal Considerations

Self-heating degrades the temperature measurement accuracy of the MAX6675 in some applications. The magnitude of the temperature errors depends on the thermal conductivity of the MAX6675 package, the

mounting technique, and the effects of airflow. Use a large ground plane to improve the temperature measurement accuracy of the MAX6675.

The accuracy of a thermocouple system can also be improved by following these precautions:

- Use the largest wire possible that does not shunt heat away from the measurement area.
- If small wire is required, use it only in the region of the measurement and use extension wire for the region with no temperature gradient.
- Avoid mechanical stress and vibration, which could strain the wires.
- When using long thermocouple wires, use a twisted-pair extension wire.
- Avoid steep temperature gradients.
- Try to use the thermocouple wire well within its temperature rating.
- Use the proper sheathing material in hostile environments to protect the thermocouple wire.
- Use extension wire only at low temperatures and only in regions of small gradients.
- Keep an event log and a continuous record of thermocouple resistance.

Reducing Effects of Pick-Up Noise

The input amplifier (A1) is a low-noise amplifier designed to enable high-precision input sensing. Keep the thermocouple and connecting wires away from electrical noise sources.

Chip Information

TRANSISTOR COUNT: 6720

PROCESS: BiCMOS

Cold-Junction-Compensated K-Thermocouple-to-Digital Converter (0°C to +1024°C)

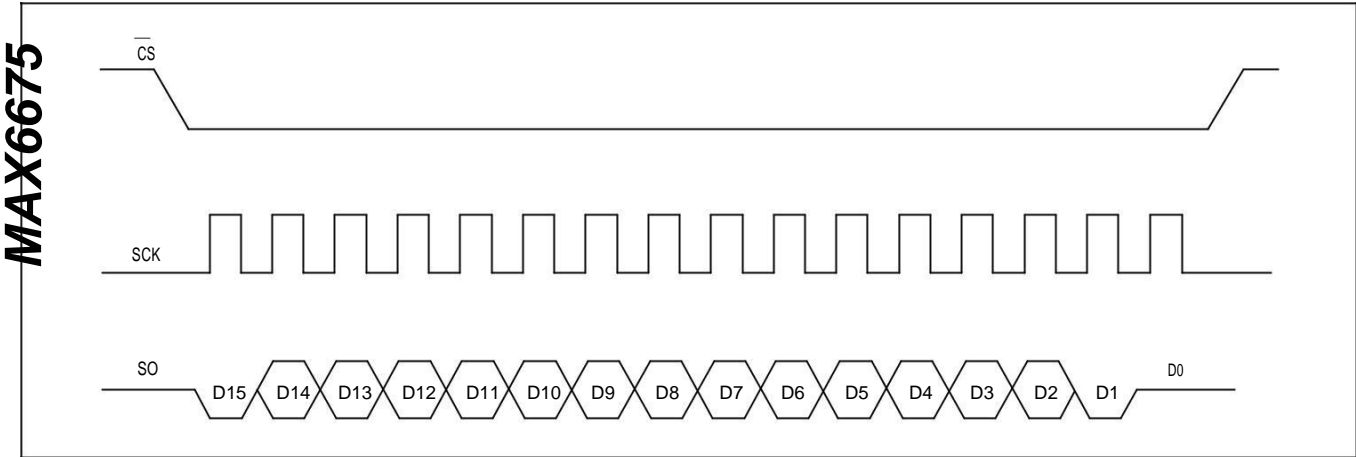


Figure 1a. Serial Interface Protocol

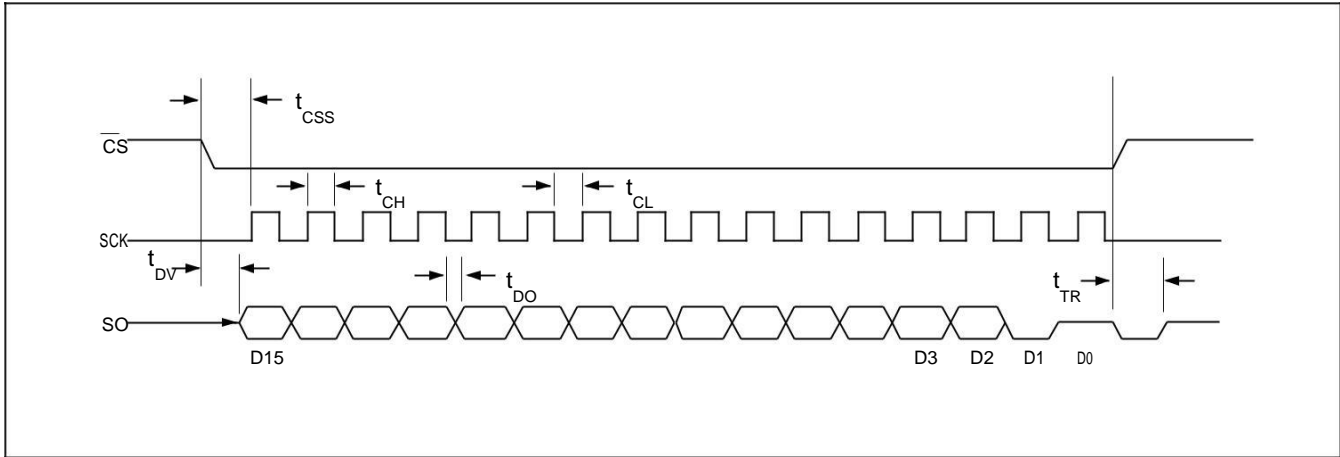


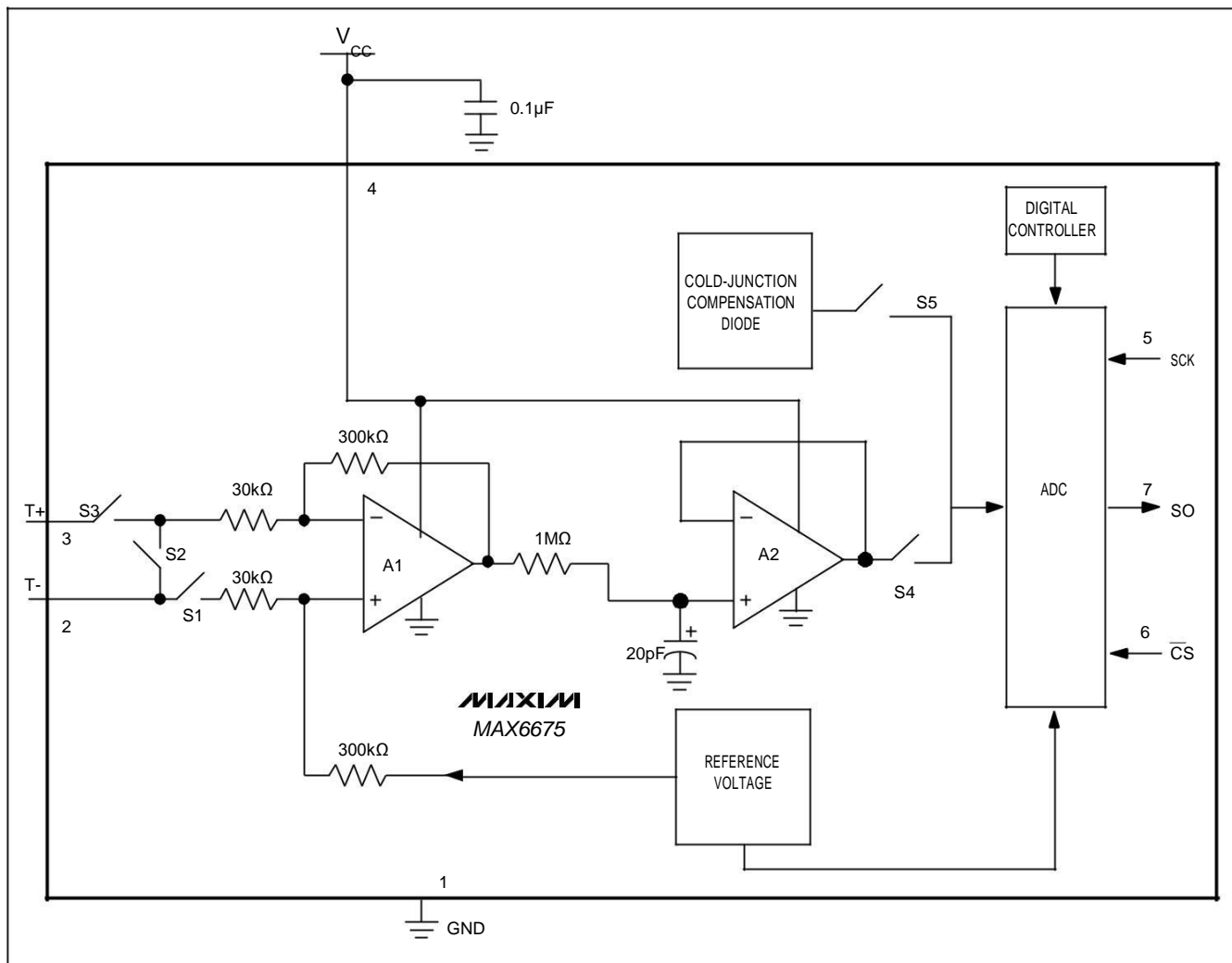
Figure 1b. Serial Interface Timing

BIT	DUMMY SIGN BIT	12-BIT TEMPERATURE READING												THERMOCOUPLE INPUT	DEVICE ID	STATE
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	0	MSB											LSB		0	Three- state

Figure 2. SO Output

Cold-Junction-Compensated K-Thermocouple-to-Digital Converter (0°C to +1024°C)

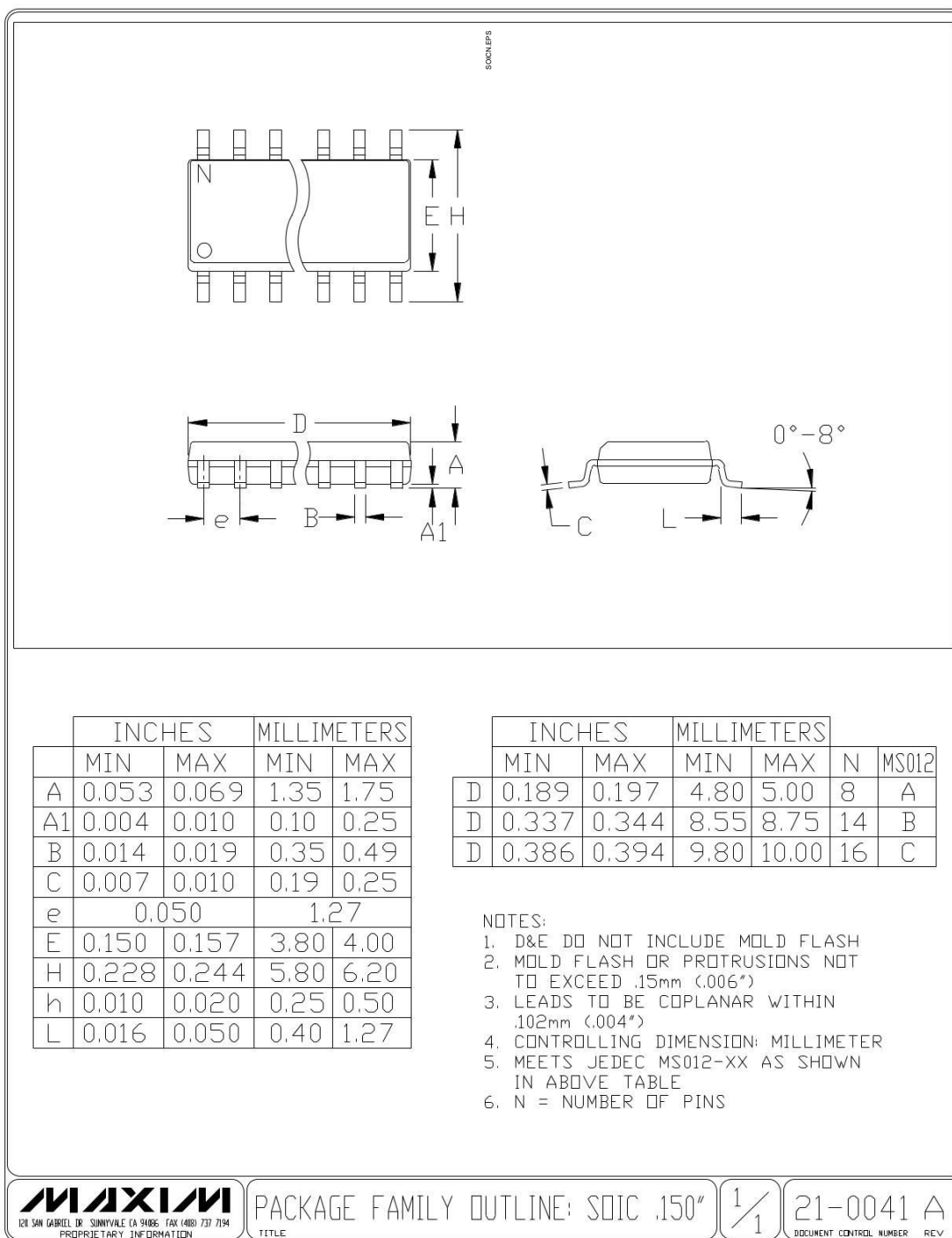
Block Diagram



MAX6675

Cold-Junction-Compensated K-Thermocouple-to-Digital Converter (0°C to +1024°C)

Package Information



Maxim cannot assume responsibility for use of any circuitry other than circuitry entirely embodied in a Maxim product. No circuit patent licenses are implied. Maxim reserves the right to change the circuitry and specifications without notice at any time.

8 **Maxim Integrated Products, 120 San Gabriel Drive, Sunnyvale, CA 94086 408-737-7600**

TABLE 9 Type K Thermocouple— thermoelectric voltage as a function of temperature (°C); reference junctions at 0 °C

°C	0	1	2	3	4	5	6	7	8	9	10	°C
Thermoelectric Voltage in Millivolts												
-270	-6.458											-270
-260	-6.411	-6.444	-6.446	-6.448	-6.450	-6.452	-6.453	-6.455	-6.456	-6.457	-6.458	-260
-250	-6.404	-6.408	-6.413	-6.417	-6.421	-6.425	-6.429	-6.432	-6.435	-6.438	-6.441	-250
-240	-6.344	-6.351	-6.358	-6.364	-6.370	-6.377	-6.382	-6.388	-6.393	-6.399	-6.404	-240
-230	-6.262	-6.271	-6.280	-6.289	-6.297	-6.306	-6.314	-6.322	-6.329	-6.337	-6.344	-230
-220	-6.158	-6.170	-6.181	-6.192	-6.202	-6.213	-6.223	-6.233	-6.243	-6.252	-6.262	-220
-210	-6.035	-6.048	-6.061	-6.074	-6.087	-6.099	-6.111	-6.123	-6.135	-6.147	-6.158	-210
-200	-5.891	-5.907	-5.922	-5.936	-5.951	-5.965	-5.980	-5.994	-6.007	-6.021	-6.035	-200
-190	-5.730	-5.747	-5.763	-5.780	-5.797	-5.813	-5.829	-5.845	-5.861	-5.876	-5.891	-190
-180	-5.550	-5.569	-5.588	-5.606	-5.624	-5.642	-5.660	-5.678	-5.695	-5.713	-5.730	-180
-170	-5.354	-5.374	-5.395	-5.415	-5.435	-5.454	-5.474	-5.493	-5.512	-5.531	-5.550	-170
-160	-5.141	-5.163	-5.185	-5.207	-5.228	-5.250	-5.271	-5.292	-5.313	-5.333	-5.354	-160
-150	-4.913	-4.936	-4.960	-4.983	-5.006	-5.029	-5.052	-5.074	-5.097	-5.119	-5.141	-150
-140	-4.669	-4.694	-4.719	-4.744	-4.768	-4.793	-4.817	-4.841	-4.865	-4.889	-4.913	-140
-130	-4.411	-4.437	-4.463	-4.490	-4.516	-4.542	-4.567	-4.593	-4.618	-4.644	-4.669	-130
-120	-4.138	-4.166	-4.194	-4.221	-4.249	-4.276	-4.303	-4.330	-4.357	-4.384	-4.411	-120
-110	-3.852	-3.882	-3.911	-3.939	-3.968	-3.997	-4.025	-4.054	-4.082	-4.110	-4.138	-110
-100	-3.554	-3.584	-3.614	-3.645	-3.675	-3.705	-3.734	-3.764	-3.794	-3.823	-3.852	-100
-90	-3.243	-3.274	-3.306	-3.337	-3.368	-3.400	-3.431	-3.462	-3.492	-3.523	-3.554	-90
-80	-2.920	-2.953	-2.986	-3.018	-3.050	-3.083	-3.115	-3.147	-3.179	-3.211	-3.243	-80
-70	-2.587	-2.620	-2.654	-2.688	-2.721	-2.755	-2.788	-2.821	-2.854	-2.887	-2.920	-70
-60	-2.243	-2.278	-2.312	-2.347	-2.382	-2.416	-2.450	-2.485	-2.519	-2.553	-2.587	-60
-50	-1.889	-1.925	-1.961	-1.996	-2.032	-2.067	-2.103	-2.138	-2.173	-2.208	-2.243	-50
-40	-1.527	-1.564	-1.600	-1.637	-1.673	-1.709	-1.745	-1.782	-1.818	-1.854	-1.889	-40
-30	-1.156	-1.194	-1.231	-1.268	-1.305	-1.343	-1.380	-1.417	-1.453	-1.490	-1.527	-30
-20	-0.778	-0.816	-0.854	-0.892	-0.930	-0.968	-1.006	-1.043	-1.081	-1.119	-1.156	-20
-10	-0.392	-0.431	-0.470	-0.508	-0.547	-0.586	-0.624	-0.663	-0.701	-0.739	-0.778	-10
0	0.000	-0.039	-0.079	-0.118	-0.157	-0.197	-0.236	-0.275	-0.314	-0.353	-0.392	0
0	0.000	0.039	0.079	0.119	0.158	0.198	0.238	0.277	0.317	0.357	0.397	0
10	0.397	0.437	0.477	0.517	0.557	0.597	0.637	0.677	0.718	0.758	0.798	10
20	0.798	0.838	0.879	0.919	0.960	1.000	1.041	1.081	1.122	1.163	1.203	20
30	1.203	1.244	1.285	1.326	1.366	1.407	1.448	1.489	1.530	1.571	1.612	30
40	1.612	1.653	1.694	1.735	1.776	1.817	1.858	1.899	1.941	1.982	2.023	40
50	2.023	2.064	2.106	2.147	2.188	2.230	2.271	2.312	2.354	2.395	2.436	50
60	2.436	2.478	2.519	2.561	2.602	2.644	2.685	2.727	2.768	2.810	2.851	60
70	2.851	2.893	2.934	2.976	3.017	3.059	3.100	3.142	3.184	3.225	3.267	70
80	3.267	3.308	3.350	3.391	3.433	3.474	3.516	3.557	3.599	3.640	3.682	80
90	3.682	3.723	3.765	3.806	3.848	3.889	3.931	3.972	4.013	4.055	4.096	90
100	4.096	4.138	4.179	4.220	4.262	4.303	4.344	4.385	4.427	4.468	4.509	100
110	4.509	4.550	4.591	4.633	4.674	4.715	4.756	4.797	4.838	4.879	4.920	110
120	4.920	4.961	5.002	5.043	5.084	5.124	5.165	5.206	5.247	5.288	5.328	120
130	5.328	5.369	5.410	5.450	5.491	5.532	5.572	5.613	5.653	5.694	5.735	130
140	5.735	5.775	5.815	5.856	5.896	5.937	5.977	6.017	6.058	6.098	6.138	140
150	6.138	6.179	6.219	6.259	6.299	6.339	6.380	6.420	6.460	6.500	6.540	150
160	6.540	6.580	6.620	6.660	6.701	6.741	6.781	6.821	6.861	6.901	6.941	160
170	6.941	6.981	7.021	7.060	7.100	7.140	7.180	7.220	7.260	7.300	7.340	170
180	7.340	7.380	7.420	7.460	7.500	7.540	7.579	7.619	7.659	7.699	7.739	180
190	7.739	7.779	7.819	7.859	7.899	7.939	7.979	8.019	8.059	8.099	8.138	190

TABLE 9 Type K Thermocouple— thermoelectric voltage as a function of temperature (°C); reference junctions at 0 °C

K°C

°C	0	1	2	3	4	5	6	7	8	9	10	°C
Thermoelectric Voltage in Millivolts												
200	8.138	8.178	8.218	8.258	8.298	8.338	8.378	8.418	8.458	8.499	8.539	200
210	8.539	8.579	8.619	8.659	8.699	8.739	8.779	8.819	8.860	8.900	8.940	210
220	8.940	8.980	9.020	9.061	9.101	9.141	9.181	9.222	9.262	9.302	9.343	220
230	9.343	9.383	9.423	9.464	9.504	9.545	9.585	9.626	9.666	9.707	9.747	230
240	9.747	9.788	9.828	9.869	9.909	9.950	9.991	10.031	10.072	10.113	10.153	240
250	10.153	10.194	10.235	10.276	10.316	10.357	10.398	10.439	10.480	10.520	10.561	250
260	10.561	10.602	10.643	10.684	10.725	10.766	10.807	10.848	10.889	10.930	10.971	260
270	10.971	11.012	11.053	11.094	11.135	11.176	11.217	11.259	11.300	11.341	11.382	270
280	11.382	11.423	11.465	11.506	11.547	11.588	11.630	11.671	11.712	11.753	11.795	280
290	11.795	11.836	11.877	11.919	11.960	12.001	12.043	12.084	12.126	12.167	12.209	290
300	12.209	12.250	12.291	12.333	12.374	12.416	12.457	12.499	12.540	12.582	12.624	300
310	12.624	12.665	12.707	12.748	12.790	12.831	12.873	12.915	12.956	12.998	13.040	310
320	13.040	13.081	13.123	13.165	13.206	13.248	13.290	13.331	13.373	13.415	13.457	320
330	13.457	13.498	13.540	13.582	13.624	13.665	13.707	13.749	13.791	13.833	13.874	330
340	13.874	13.916	13.958	14.000	14.042	14.084	14.126	14.167	14.209	14.251	14.293	340
350	14.293	14.335	14.377	14.419	14.461	14.503	14.545	14.587	14.629	14.671	14.713	350
360	14.713	14.755	14.797	14.839	14.881	14.923	14.965	15.007	15.049	15.091	15.133	360
370	15.133	15.175	15.217	15.259	15.301	15.343	15.385	15.427	15.469	15.511	15.554	370
380	15.554	15.596	15.638	15.680	15.722	15.764	15.806	15.849	15.891	15.933	15.975	380
390	15.975	16.017	16.059	16.102	16.144	16.186	16.228	16.270	16.313	16.355	16.397	390
400	16.397	16.439	16.482	16.524	16.566	16.608	16.651	16.693	16.735	16.778	16.820	400
410	16.820	16.862	16.904	16.947	16.989	17.031	17.074	17.116	17.158	17.201	17.243	410
420	17.243	17.285	17.328	17.370	17.413	17.455	17.497	17.540	17.582	17.624	17.667	420
430	17.667	17.709	17.752	17.794	17.837	17.879	17.921	17.964	18.006	18.049	18.091	430
440	18.091	18.134	18.176	18.218	18.261	18.303	18.346	18.388	18.431	18.473	18.516	440
450	18.516	18.558	18.601	18.643	18.686	18.728	18.771	18.813	18.856	18.898	18.941	450
460	18.941	18.983	19.026	19.068	19.111	19.154	19.196	19.239	19.281	19.324	19.366	460
470	19.366	19.409	19.451	19.494	19.537	19.579	19.622	19.664	19.707	19.750	19.792	470
480	19.792	19.835	19.877	19.920	19.962	20.005	20.048	20.090	20.133	20.175	20.218	480
490	20.218	20.261	20.303	20.346	20.389	20.431	20.474	20.516	20.559	20.602	20.644	490
500	20.644	20.687	20.730	20.772	20.815	20.857	20.900	20.943	20.985	21.028	21.071	500
510	21.071	21.113	21.156	21.199	21.241	21.284	21.326	21.369	21.412	21.454	21.497	510
520	21.497	21.540	21.582	21.625	21.668	21.710	21.753	21.796	21.838	21.881	21.924	520
530	21.924	21.966	22.009	22.052	22.094	22.137	22.179	22.222	22.265	22.307	22.350	530
540	22.350	22.393	22.435	22.478	22.521	22.563	22.606	22.649	22.691	22.734	22.776	540
550	22.776	22.819	22.862	22.904	22.947	22.990	23.032	23.075	23.117	23.160	23.203	550
560	23.203	23.245	23.288	23.331	23.373	23.416	23.458	23.501	23.544	23.586	23.629	560
570	23.629	23.671	23.714	23.757	23.799	23.842	23.884	23.927	23.970	24.012	24.055	570
580	24.055	24.097	24.140	24.182	24.225	24.267	24.310	24.353	24.395	24.438	24.480	580
590	24.480	24.523	24.565	24.608	24.650	24.693	24.735	24.778	24.820	24.863	24.905	590
600	24.905	24.948	24.990	25.033	25.075	25.118	25.160	25.203	25.245	25.288	25.330	600
610	25.330	25.373	25.415	25.458	25.500	25.543	25.585	25.627	25.670	25.712	25.755	610
620	25.755	25.797	25.840	25.882	25.924	25.967	26.009	26.052	26.094	26.136	26.179	620
630	26.179	26.221	26.263	26.306	26.348	26.390	26.433	26.475	26.517	26.560	26.602	630
640	26.602	26.644	26.687	26.729	26.771	26.814	26.856	26.898	26.940	26.983	27.025	640
650	27.025	27.067	27.109	27.152	27.194	27.236	27.278	27.320	27.363	27.405	27.447	650
660	27.447	27.489	27.531	27.574	27.616	27.658	27.700	27.742	27.784	27.826	27.869	660
670	27.869	27.911	27.953	27.995	28.037	28.079	28.121	28.163	28.205	28.247	28.289	670
680	28.289	28.332	28.374	28.416	28.458	28.500	28.542	28.584	28.626	28.668	28.710	680
690	28.710	28.752	28.794	28.835	28.877	28.919	28.961	29.003	29.045	29.087	29.129	690
°C	0	1	2	3	4	5	6	7	8	9	10	°C

TABLE 9 Type K Thermocouple— thermoelectric voltage as a function of temperature (°C); reference junctions at 0 °C

°C	0	1	2	3	4	5	6	7	8	9	10	°C
Thermoelectric Voltage in Millivolts												
700	29.129	29.171	29.213	29.255	29.297	29.338	29.380	29.422	29.464	29.506	29.548	700
710	29.548	29.589	29.631	29.673	29.715	29.757	29.798	29.840	29.882	29.924	29.965	710
720	29.965	30.007	30.049	30.090	30.132	30.174	30.216	30.257	30.299	30.341	30.382	720
730	30.382	30.424	30.466	30.507	30.549	30.590	30.632	30.674	30.715	30.757	30.798	730
740	30.798	30.840	30.881	30.923	30.964	31.006	31.047	31.089	31.130	31.172	31.213	740
750	31.213	31.255	31.296	31.338	31.379	31.421	31.462	31.504	31.545	31.586	31.628	750
760	31.628	31.669	31.710	31.752	31.793	31.834	31.876	31.917	31.958	32.000	32.041	760
770	32.041	32.082	32.124	32.165	32.206	32.247	32.289	32.330	32.371	32.412	32.453	770
780	32.453	32.495	32.536	32.577	32.618	32.659	32.700	32.742	32.783	32.824	32.865	780
790	32.865	32.906	32.947	32.988	33.029	33.070	33.111	33.152	33.193	33.234	33.275	790
800	33.275	33.316	33.357	33.398	33.439	33.480	33.521	33.562	33.603	33.644	33.685	800
810	33.685	33.726	33.767	33.808	33.848	33.889	33.930	33.971	34.012	34.053	34.093	810
820	34.093	34.134	34.175	34.216	34.257	34.297	34.338	34.379	34.420	34.460	34.501	820
830	34.501	34.542	34.582	34.623	34.664	34.704	34.745	34.786	34.826	34.867	34.908	830
840	34.908	34.948	34.989	35.029	35.070	35.110	35.151	35.192	35.232	35.273	35.313	840
850	35.313	35.354	35.394	35.435	35.475	35.516	35.556	35.596	35.637	35.677	35.718	850
860	35.718	35.758	35.798	35.839	35.879	35.920	35.960	36.000	36.041	36.081	36.121	860
870	36.121	36.162	36.202	36.242	36.282	36.323	36.363	36.403	36.443	36.484	36.524	870
880	36.524	36.564	36.604	36.644	36.685	36.725	36.765	36.805	36.845	36.885	36.925	880
890	36.925	36.965	37.006	37.046	37.086	37.126	37.166	37.206	37.246	37.286	37.326	890
900	37.326	37.366	37.406	37.446	37.486	37.526	37.566	37.606	37.646	37.686	37.725	900
910	37.725	37.765	37.805	37.845	37.885	37.925	37.965	38.005	38.044	38.084	38.124	910
920	38.124	38.164	38.204	38.243	38.283	38.323	38.363	38.402	38.442	38.482	38.522	920
930	38.522	38.561	38.601	38.641	38.680	38.720	38.760	38.799	38.839	38.878	38.918	930
940	38.918	38.958	38.997	39.037	39.076	39.116	39.155	39.195	39.235	39.274	39.314	940
950	39.314	39.353	39.393	39.432	39.471	39.511	39.550	39.590	39.629	39.669	39.708	950
960	39.708	39.747	39.787	39.826	39.866	39.905	39.944	39.984	40.023	40.062	40.101	960
970	40.101	40.141	40.180	40.219	40.259	40.298	40.337	40.376	40.415	40.455	40.494	970
980	40.494	40.533	40.572	40.611	40.651	40.690	40.729	40.768	40.807	40.846	40.885	980
990	40.885	40.924	40.963	41.002	41.042	41.081	41.120	41.159	41.198	41.237	41.276	990
1000	41.276	41.315	41.354	41.393	41.431	41.470	41.509	41.548	41.587	41.626	41.665	1000
1010	41.665	41.704	41.743	41.781	41.820	41.859	41.898	41.937	41.976	42.014	42.053	1010
1020	42.053	42.092	42.131	42.169	42.208	42.247	42.286	42.324	42.363	42.402	42.440	1020
1030	42.440	42.479	42.518	42.556	42.595	42.633	42.672	42.711	42.749	42.788	42.826	1030
1040	42.826	42.865	42.903	42.942	42.980	43.019	43.057	43.096	43.134	43.173	43.211	1040
1050	43.211	43.250	43.288	43.327	43.365	43.403	43.442	43.480	43.518	43.557	43.595	1050
1060	43.595	43.633	43.672	43.710	43.748	43.787	43.825	43.863	43.901	43.940	43.978	1060
1070	43.978	44.016	44.054	44.092	44.130	44.169	44.207	44.245	44.283	44.321	44.359	1070
1080	44.359	44.397	44.435	44.473	44.512	44.550	44.588	44.626	44.664	44.702	44.740	1080
1090	44.740	44.778	44.816	44.853	44.891	44.929	44.967	45.005	45.043	45.081	45.119	1090
1100	45.119	45.157	45.194	45.232	45.270	45.308	45.346	45.383	45.421	45.459	45.497	1100
1110	45.497	45.534	45.572	45.610	45.647	45.685	45.723	45.760	45.798	45.836	45.873	1110
1120	45.873	45.911	45.948	45.986	46.024	46.061	46.099	46.136	46.174	46.211	46.249	1120
1130	46.249	46.286	46.324	46.361	46.398	46.436	46.473	46.511	46.548	46.585	46.623	1130
1140	46.623	46.660	46.697	46.735	46.772	46.809	46.847	46.884	46.921	46.958	46.995	1140
1150	46.995	47.033	47.070	47.107	47.144	47.181	47.218	47.256	47.293	47.330	47.367	1150
1160	47.367	47.404	47.441	47.478	47.515	47.552	47.589	47.626	47.663	47.700	47.737	1160
1170	47.737	47.774	47.811	47.848	47.884	47.921	47.958	47.995	48.032	48.069	48.105	1170
1180	48.105	48.142	48.179	48.216	48.252	48.289	48.326	48.363	48.399	48.436	48.473	1180
1190	48.473	48.509	48.546	48.582	48.619	48.656	48.692	48.729	48.765	48.802	48.838	1190
°C	0	1	2	3	4	5	6	7	8	9	10	°C

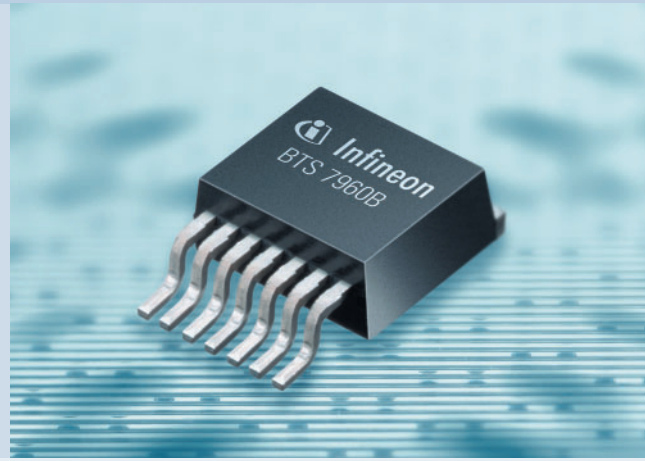
TABLE 9 Type K Thermocouple— thermoelectric voltage as a function of temperature (°C); reference junctions at 0 °C

K°C

°C	0	1	2	3	4	5	6	7	8	9	10	°C
Thermoelectric Voltage in Millivolts												
1200	48.838	48.875	48.911	48.948	48.984	49.021	49.057	49.093	49.130	49.166	49.202	1200
1210	49.202	49.239	49.275	49.311	49.348	49.384	49.420	49.456	49.493	49.529	49.565	1210
1220	49.565	49.601	49.637	49.674	49.710	49.746	49.782	49.818	49.854	49.890	49.926	1220
1230	49.926	49.962	49.998	50.034	50.070	50.106	50.142	50.178	50.214	50.250	50.286	1230
1240	50.286	50.322	50.358	50.393	50.429	50.465	50.501	50.537	50.572	50.608	50.644	1240
1250	50.644	50.680	50.715	50.751	50.787	50.822	50.858	50.894	50.929	50.965	51.000	1250
1260	51.000	51.036	51.071	51.107	51.142	51.178	51.213	51.249	51.284	51.320	51.355	1260
1270	51.355	51.391	51.426	51.461	51.497	51.532	51.567	51.603	51.638	51.673	51.708	1270
1280	51.708	51.744	51.779	51.814	51.849	51.885	51.920	51.955	51.990	52.025	52.060	1280
1290	52.060	52.095	52.130	52.165	52.200	52.235	52.270	52.305	52.340	52.375	52.410	1290
1300	52.410	52.445	52.480	52.515	52.550	52.585	52.620	52.654	52.689	52.724	52.759	1300
1310	52.759	52.794	52.828	52.863	52.898	52.932	52.967	53.002	53.037	53.071	53.106	1310
1320	53.106	53.140	53.175	53.210	53.244	53.279	53.313	53.348	53.382	53.417	53.451	1320
1330	53.451	53.486	53.520	53.555	53.589	53.623	53.658	53.692	53.727	53.761	53.795	1330
1340	53.795	53.830	53.864	53.898	53.932	53.967	54.001	54.035	54.069	54.104	54.138	1340
1350	54.138	54.172	54.206	54.240	54.274	54.308	54.343	54.377	54.411	54.445	54.479	1350
1360	54.479	54.513	54.547	54.581	54.615	54.649	54.683	54.717	54.751	54.785	54.819	1360
1370	54.819	54.852	54.886									1370

BTS 7960B

High Current PN Half Bridge
NovalithIC™



THE BTS 7960B is a fully integrated high current half bridge for motor drive applications. It is part of the NovalithIC™ family containing one p-channel highside MOSFET and one n-channel lowside MOSFET with an integrated driver IC in one package. The power switches utilize vertical MOS technologies to ensure optimum on state resistance. Due to the p-channel highside switch the need for a charge pump is eliminated thus minimizing EMI. Interfacing to a microcontroller is made easy by the integrated driver IC which features logic level inputs, diagnosis with current sense, slew rate adjustment, dead time generation and protection against overtemperature, overvoltage, undervoltage, overcurrent and short circuit.

THE BTS 7960B provides a cost optimized solution for protected high current PWM motor drives with very low board space consumption.

Applications

High current DC- or brushless DC (BLDC) motor drives for use in applications like

- Power Windows
- Sun Roof
- Central Door Lock
- Seat Positioning
- Wiper
- Washer Pump
- 4 Wheel Transfer Case
- Electronically Controlled Manual Transmission (ECMT)
- Cooling Fan

Features

- High peak current capability of min. 33 A
- Path resistance of typ. 16 mΩ@ 25°C
- Low quiescent current of typ. 7 μA@ 25°C
- PWM capability of up to 25 kHz with active freewheeling
- Switched mode current limitation for reduced power dissipation in overcurrent
- Status flag diagnosis with current sense capability
- Overtemperature shut down with latch behaviour
- Overvoltage and undervoltage shut down
- Driver circuit with logic level inputs
- Adjustable slew rates for optimized EMI

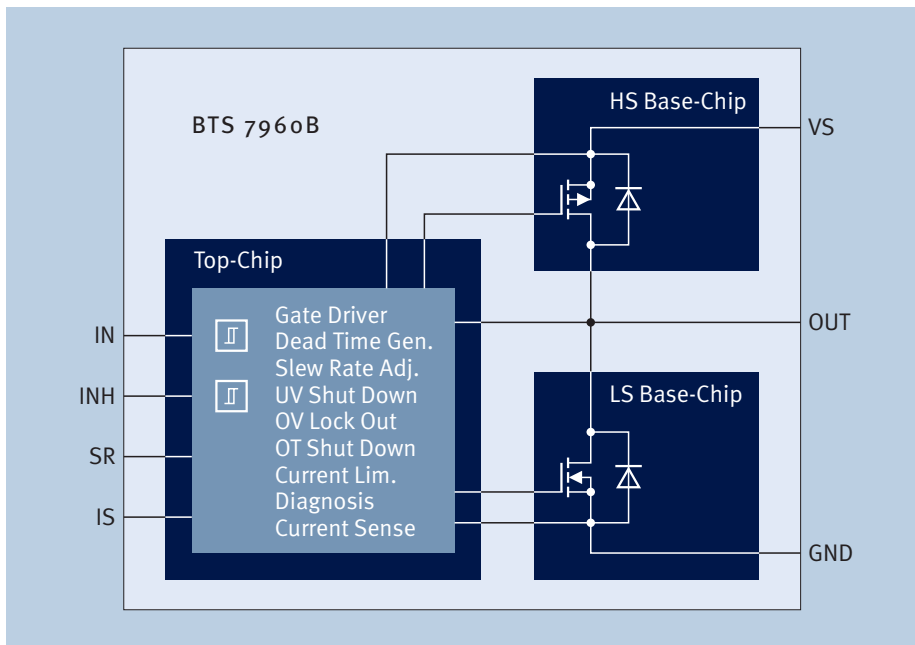
www.infineon.com/bridges

Automotive Power



Never stop thinking.

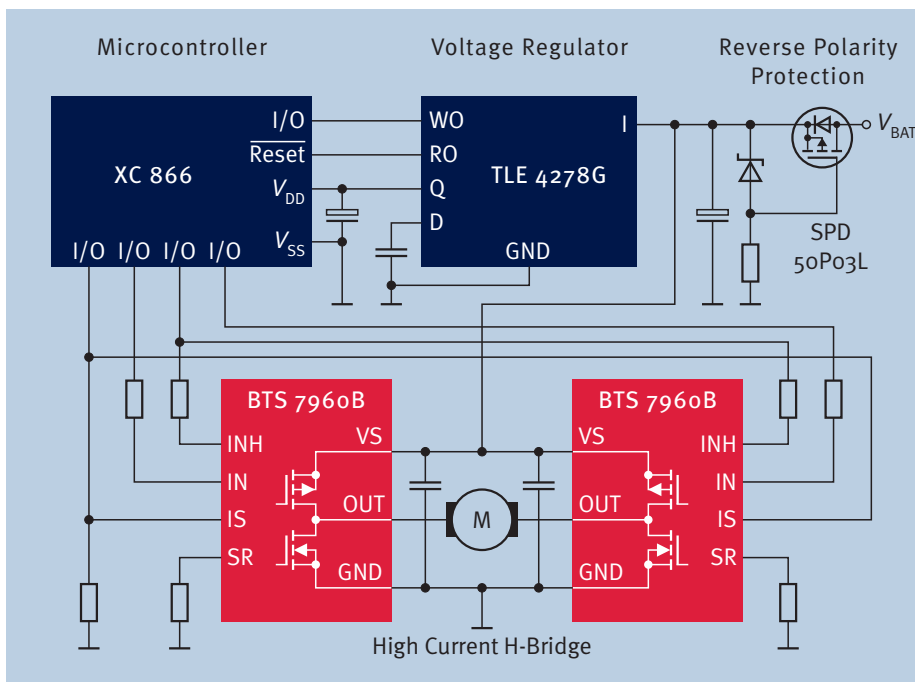
Block Diagram



Product Summary

Type	Sales Code	Description	Package
BTS 7960B	Q67060-S6160	High Current PN Half Bridge	P-TO263-7-6

Application Example



How to reach us:
<http://www.infineon.com>

Published by
 Infineon Technologies AG
 St.-Martin-Strasse 53
 81669 München

© Infineon Technologies AG 17. August 2005.
 All Rights Reserved.

Template: pb_tmplt.fm/4

Attention please!

The information herein is given to describe certain components and shall not be considered as a guarantee of characteristics. Terms of delivery and rights to technical change reserved. We hereby disclaim any and all warranties, including but not limited to warranties of non-infringement, regarding circuits, descriptions and charts stated herein.

Information

For further information on technology, delivery terms and conditions and prices please contact your nearest Infineon Technologies Office.

Warnings

Due to technical requirements components may contain dangerous substances. For information on the types in question please contact your nearest Infineon Technologies Office.

Infineon Technologies Components may only be used in life-support devices or systems with the express written approval of Infineon Technologies, if a failure of such components can reasonably be expected to cause the failure of that life-support device or system, or to affect the safety or effectiveness of that device or system. Life support devices or systems are intended to be implanted in the human body, or to support and/or maintain and sustain and/or protect human life. If they fail, it is reasonable to assume that the health of the user or other persons may be endangered.

Ordering No. B152-H8512-G1-X-7600
 Printed in Germany
 PS 08051. nb